Univerza v Ljubljani
Fakulteta za računalništvo in informatiko

Matic Tkalec

# Zanesljivostna analiza vzorčnega socio - tehničnega sistema na podlagi FRAM metode

DIPLOMSKO DELO
UNIVERZITETNI ŠTUDIJSKI PROGRAM PRVE STOPNJE
RAČUNALNIŠTVO IN INFORMATIKA

prof. dr. Miha Mraz
MENTOR

Ljubljana, 2017

**Tematika naloge:**

*Kandidat naj v svojem delu predstavi osnove FRAM metode in slednjo uporabi pri zanesljivostni analizi vzorčnega socio - tehničnega sistema, ki ga predstavlja izvajanje pilotovih nalog pri vzletu motornega športnega letala. Pri tem naj se kandidat osredotoči na identifikacijo sistemskih funkcij, njihovo deklaracijo in možnosti njihove resonance v skladu s FRAM metodo.*

# IZJAVA O AVTORSTVU DIPLOMSKEGA DELA

Spodaj podpisani izjavljam, da sem avtor dela, da slednje ne vsebuje materiala, ki bi ga kdorkoli predhodno že objavil ali oddal v obravnavo za pridobitev naziva na univerzi ali drugem visokošolskem zavodu, razen v primerih kjer so navedeni viri.
S svojim podpisom zagotavljam, da:

- sem delo izdelal samostojno pod mentorstvom prof. dr. Mihe Mraza,
- so elektronska oblika dela, naslov (slov., angl.), povzetek (slov., angl.) ter ključne besede (slov., angl.) identični s tiskano obliko in
- soglašam z javno objavo elektronske oblike dela v zbirki "Dela FRI".

— Matic Tkalec, Ljubljana, september 2017.

Univerza v Ljubljani
Fakulteta za računalništvo in informatiko

Matic Tkalec

# Zanesljivostna analiza vzorčnega socio - tehničnega sistema na podlagi FRAM metode

## POVZETEK

Današnji socio - tehnični sistemi postajajo vse bolj kompleksni. Potrebujemo torej sofisticiran način, s katerim bomo sposobni takšne sisteme razumeti in jih podrobno analizirati. Pojavljajo se nove metode, ki so zasnovane posebej za analizo tovrstnih sistemov, vendar so te metode še v fazi razvoja. V pričujočem delu se osredotočimo na analizo vzorčnega socio - tehničnega sistema, ki ga predstavlja izvajanje nalog pilota pri upravljanju enomotornega športnega letala. To analizo izvedemo s pomočjo FRAM metode. Najprej delovanje danega sistema podrobno opišemo, v nadaljevanju pa predstavimo uporabo FRAM metode, ki jo nato apliciramo na opisan sistem. K razvoju FRAM metode torej prispevamo tako, da proizvedemo primer aplikacije FRAM metode na konkreten kompleksen socio - tehnični sistem in poizkusimo identificirati nov analitični korak metode, ki bi to metodo pripeljal bližje k simulaciji situacij z računalniškim modelom.

**Ključne besede:** socio - tehnični sistemi, FRAM, funkcijska resonanca, sistemska analiza, splošno letalstvo

University of Ljubljana
Faculty of Computer and Information Science

Matic Tkalec

# Reliability analysis of a socio - technical system example based on the FRAM method

## ABSTRACT

Modern socio - technical systems are becoming more and more complex, thus we are in need of a sophisticated method which will enable us to understand and analyse such systems in detail. New methods are emerging for this particular purpose, yet they are presently still developing. This work focuses on analysing an example of a socio - technical system. This system is based on the tasks that a pilot of a small, general aviation aircraft has to accomplish to successfully perform flight operations. The analysis of the system is conducted with the FRAM method. At first we describe how the observed system works, then we apply the FRAM to this description. This work contributes to the development of the FRAM because it provides an example of an application of the method to a complex socio - technical system. This work also includes an attempt to identify a new analytical step, which would bring this method closer to a computer simulation of different system situations.

**Key words:** socio - technical systems, FRAM, functional resonance, systems analysis, general aviation

# ZAHVALA

*Zahvaljujem se mentorju, prof. dr. Mihi Mrazu, za vse vsebinske usmeritve pri pisanju diplomske naloge, ter za spodbudo in hitro odzivnost.*

*Iz srca bi se rad zahvalil svoji družini, tako staršem kot tudi starim staršem, za vso pomoč, podporo, potrpežljivost, vztrajnost in spodbudo v času študija. Iskrena zahvala gre tudi mojemu dekletu, za vse nasvete in spodbudne besede tekom celotnega študija.*

— Matic Tkalec, Ljubljana, september 2017.

# KAZALO

# 1 Uvod

Že vse od začetka industrijske revolucije v 18. stoletju človeštvo stremi k implementaciji raznovrstnih tehnik oziroma tehnologij v vse aspekte naših življenj. Posledično se ljudje na tehniko čedalje bolj zanašamo. Ta nas spremlja na vsakem koraku - ljudje se s tehniko srečujemo doma, v šolah, bolnicah, vrtcih, v avtomobilih itd., na vsakodnevni bazi. Na tehniko se zanašajo tudi različni transportni sistemi, tako ladijski, cestni in železniški, kot tudi letalski. Vsi našteti sistemi za pravilno delovanje potrebujejo različne tehnične komponente. Brez teh komponent ti sistemi ne bi delovali, vendar tudi zgolj te komponente navadno niso dovolj za delovanje nekega kompleksnega sistema. Tehnika je v večini primerov namreč zgolj orodje, katerega uporabljajo ljudje oziroma upravljalci teh kompleksnih sistemov. Takšni sistemi, kot je pojasnjeno na začetku poglavja 2, se imenujejo **socio - tehnični sistemi** [1]. Že sam pojem pove, da so ti sistemi v grobem razdeljeni na dva dela in sicer na socialni, človeški del sistema in na drugi, torej tehnični del sistema. Ker je teh sistemov čedalje več, je čedalje večja tudi potreba po podrobnem poznavanju tovrstnih sistemov. Ker se tako zelo zanašamo na te sisteme, jih moramo znati tudi dobro analizirati.

V socio - tehničnih sistemih včasih pride do napake oziroma odpovedi sistema. Vprašanje je, kako te napake pojasniti tako, da zajamemo vse faktorje, ki so na kakršenkoli način pripomogli k odpovedi sistema. Sistemske napake se lahko v majhnih in enostavnih sistemih mnogokrat pojasni zgolj z enostavnim razmerjem vzrok - posledica. Kadar pa imamo opravka s kompleksnimi socio - tehničnimi sistemi, določenih dogodkov ni mogoče pojasniti s takšno enostavno razlago. Tradicionalne metode, ki razmerje vzrok - posledica uporabljajo za pojasnitev odpovedi kompleksnih socio - tehničnih sistemov so danes ovrednotene kot zastarele in neprimerne za analizo le-teh. V zadnjem času se kot odgovor na pomanjkljivosti tradicionalnih metod pojavljajo nove metode za analizo tovrstnih sistemov, ki so sposobne pojasniti tudi kompleksna razmerja med različnimi komponentami sistema. V pričujočem delu omenimo dve takšni metodi in sicer STPA metodo, ki jo je razvila Nancy Leveson, osredotočimo pa se na FRAM metodo, ki jo je razvil Erik Hollnagel. S FRAM metodo, ki za analizo socio - tehničnih sistemov uporablja kvalitativni pristop [2], v pričujočem delu tudi izvedemo poizkus analize vzorčnega socio - tehničnega sistema. Če se vrnemo k delitvi socio - tehničnih sistemov na dva dela ugotovimo, da se tehnični del sistema skozi čas **izboljšuje**. Tehnika iz dneva v dan postaja bolj napredna in bolj zmogljiva - tehnični del sistema je narejen tako, da je predvidljiv in zanesljiv [2], sistemi pa kljub temu odpovedujejo. Tehnični del sistema za nas torej ni tako zanimiv. Skozi čas človeški faktor, v primerjavi s tehnologijo, ostaja nespremenjen. Pri FRAM analizi se torej osredotočamo na socialni, človeški del opazovanega socio - tehničnega sistema. Delovni pogoji za opravljanje dela v različnih socio - tehničnih sistemih so redkokdaj idealni. Upravljalci teh sistemov svoje delo konstantno prilagajajo danim delovnim pogojem in s tem torej omogočajo pravilno delovanje sistema, obenem pa lahko te prilagoditve včasih povzročijo odpoved opazovanega sistema [2].

FRAM analiza se lahko v splošnem uporabi za dva namena, ali za analizo odpovedi socio - tehničnega sistema, ali pa za zanesljivostno analizo le-tega [2]. V pričujočem delu se lotimo zanesljivostne analize socio - tehničnega sistema, ki ga predstavlja izvajanje nalog pilota enomotornega športnega letala pri postopku vzletanja. To pomeni, da želimo identificirati čimveč potencialno tveganih scenarijev, ki bi lahko vodili do neljubih dogodkov. V opazovanem socio - tehničnem sistemu se letalo nahaja na neki točki pred vzletno stezo, pilot pa mora s tem letalom pod danimi delovnimi pogoji letalo varno spraviti v zrak. Pri tem izpolnjuje naloge kot so komunikacija z letalsko kontrolo, spreminjanje konfiguracije letala in dejansko krmiljenje letala. Te naloge v poglavju 4 močno

razčlenimo in analiziramo s pomočjo FRAM metode.

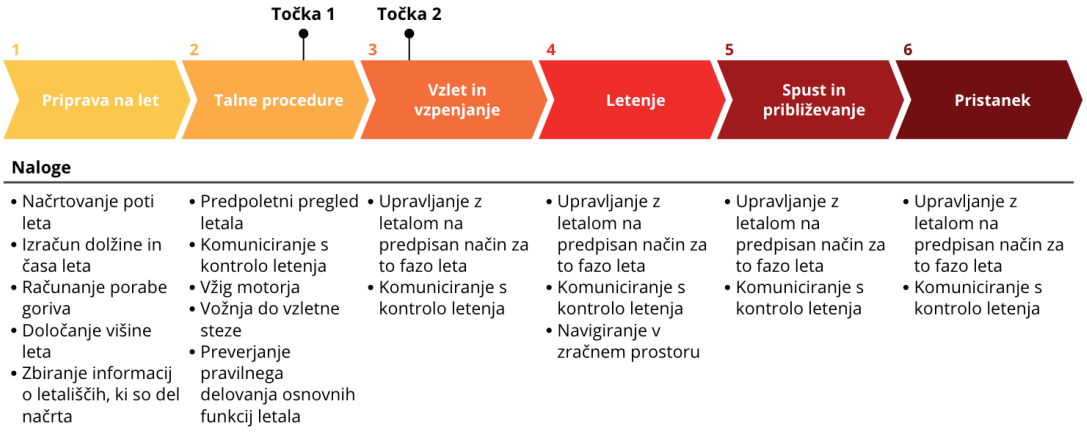
Cilj pričujočega dela je uporaba FRAM metode za zanesljivostno analizo vzorčnega socio - tehničnega sistema in identifikacija dodatnega koraka FRAM metode, ki bi lahko metodo pripeljal korak bližje k računalniški simulaciji situacij v opazovanem sistemu.

# 2 Predstavitev problematike

Danes se skoraj v vsakem delovnem okolju srečujemo s takšnimi in drugačnimi tehnološkimi rešitvami oziroma tehničnimi sistemi. Ti sistemi omogočajo opravljanje nalog v specifičnem delovnem okolju, vendar kljub svoji dovršenosti ne znajo in ne zmorejo delovati popolnoma samostojno. Tovrstni sistemi običajno potrebujejo upravljalca, človeka, ki z njimi upravlja po točno določenih pravilih. Eden izmed razlogov za potrebo po upravljalcu je, da sistem ne zna "razmišljati" sam, torej se ne zna prilagajati spreminjajočim se razmeram v delovnem okolju, oziroma je njegova sposobnost prilagajanja tem razmeram omejena. Ker pa se razmere v nekaterih delovnih okoljih lahko spreminjajo drastično in hitro, za pravilno delovanje tehničnega sistema nujno potrebujemo nekoga, ki bo dovolj hitro in uspešno prilagodil sistem trenutnim razmeram. To počne upravljalec sistema. Vsakršno ciljno orientirano dejavnost, ki za svoje delovanje uporablja skupek tehničnih in socialnih komponent, ki so druga od druge odvisne, lahko torej kategoriziramo kot **socio - tehnični sistem** [1].

## 2.1 Opis opazovanega socio - tehničnega sistema

V diplomski nalogi se osredotočimo na opis in analizo socio - tehničnega sistema (angl. *socio - technical system*) v splošnem letalstvu (angl. *general aviation*)[1]. Vživimo se v vlogo pilota manjšega letala[2] in poizkusimo najpomembnejša dogajanja, ki se odvijajo v tem letalu v času leta in pred njim, analizirati. Letalo v času izvajanja letalskih operacij klasificiramo kot socio - tehnični sistem; letalo leti zaradi svojih tehničnih lastnosti, vendar potrebuje človeka, ki z razpoložljivo tehniko upravlja. Za uspešen polet mora pilot uspešno opraviti mnogo različnih nalog. Na sliki 2.1 je prikazan časovni trak, ki glavne naloge pilota jasno opredeli in našteje nekaj njihovih podnalog. V razdelku 2.2 je vsaka izmed teh nalog bolj podrobno opisana.




**Slika 2.1** Časovni trak, na katerem je prikazano zaporedje dogodkov pri običajnem poletu (ustvarjeno z Vizzlo [3]).

## 2.2 Naloge pilota pri izvajanju letalskih operacij

Z naštevanjem in opisom nalog pilota skušamo karseda dobro zajeti velikost in kompleksnost socio - tehničnega sistema, ki smo ga opredelili kot predmet vzorčne analize. Naloge, ki so predstavljene na sliki 2.1 in jih bomo v nadaljevanju podrobneje opisali, so sledeče:

[1] Splošno letalstvo zajema ves letalski promet, ki ni linijski ali vojaški

[2] Enomotorno propelersko letalo, naprimer klasična Cessna - 172

- priprava na let (angl. *flight preparation*),
- talne procedure (angl. *ground procedures*),
- vzlet in vzpenjanje (angl. *takeoff and climb*),
- letenje (angl. *cruise*),
- spust in približevanje (angl. *descent and approach*),
- pristanek (angl. *landing*).

### 2.2.1 Priprava na let

V pripravo na let spadajo podnaloge, ki med letom pomagajo pilotu pri upravljanju letala. Pri pripravi na polet pilot najprej izdela načrt leta. Na letalski zemljevid z ravnimi črtami nariše potek leta, opredeli začetno in končno točko leta ter vse vmesne točke in določi smer letenja, čas letenja in razdaljo med sosednjimi navigacijskimi točkami. Vsi našteti podatki so praviloma napisani na letalskem zemljevidu, poleg tega pa pilot vse podatke prepiše v obrazec, namenjen navigaciji (angl. *navigation log*), s katerim si potem med potekom leta pomaga pri navigiranju. Pilot za vsak del leta določi višino, na kateri bo letel in jo vpiše v navigacijski obrazec. Izračunati mora tudi, koliko goriva bo letalo porabilo pri izvedbi leta in poskrbeti, da ima letalo pred izvedbo leta dovolj goriva, vključno z rezervnim gorivom za nujne primere. Ko ima pilot načrt leta dokončno urejen, mora tega posredovati zračni kontroli; pri izvajanju letalskih operacij v slovenskem zračnem prostoru to lahko stori tako, da izpolni spletni obrazec z informacijami o načrtovanem poletu.

Za uspešno pripravo na polet je potrebno pridobiti zelo pomembne informacije o letališčih, ki so vključena v načrt leta, kot so nadmorska višina letališča, informacije o vzletni stezi letališča, frekvenca, na kateri poteka komunikacija s kontrolnim stolpom letališča in podobno. Pilot mora tudi pridobiti informacije o morebitnih spremembah, izrednih dogodkih ali prepovedih v zračnem prostoru, v katerem namerava leteti. Ti so dosegljivi v obliki NOTAM (angl. *Notice To Airmen*) obvestil, ki so objavljena na spletu. Prav tako mora pilot vedeti v kakšnem vremenu bo letel. To lahko ugotovi iz različnih vremenskih portalov, za večja letališča pa so na spletu na voljo METAR (angl. *Meteorological Aerodrome Report*) depeše, ki bolj podrobno opišejo trenutne vremenske

razmere na specifičnem letališču. Če pilot ugotovi, da so pogoji za letenje primerni in so opravljene vse podnaloge priprave na let, se je na izvedbo leta uspešno pripravil.

### 2.2.2 Talne procedure

V okviru talnih procedur so zajete podnaloge, ki jih pilot opravi od prihoda na letališče, do samega vzleta. Pilot mora najprej pregledati letalo in pri tem posvečati posebno pozornost količini goriva v rezervoarjih, količini olja v motorju, stanju gum in podobno. Ko to opravi in ugotovi, da je letalo pripravljeno za izvedbo načrtovanega leta, se lahko usede v letalo in prične z izvajanjem letalskih operacij. V letalu se vse naloge opravljajo z uporabo *checklist*[3]. Začne torej z izvajanjem *checklist*, ki so predpisane za vžig motorja. Zatem kontaktira kontrolo letenja in jasno pove svoj namen. Ko dobi dovoljenje za vožnjo po stezi, namenjeni za vožnjo po tleh (angl. *taxiway*), spet izvede *checkliste* za ta del leta. S tem preveri, če delujejo zavore, zakrilca in ostale osnovne komponente letala. Pri vožnji po stezi se mora pilot strogo držati navodil, ki jih je prejel s strani kontrole letenja. Ko se enkrat nahaja pred vzletno stezo, izvede še zadnje *checkliste* pred izvedbo poleta. Tako lahko pilot preveri pravilno delovanje motorja in nekaterih njegovih komponent, ter se pripravi na izvedbo vzleta. Pilot mora, preden lahko nadaljuje pot na vzletno stezo, za to dejanje pridobiti dovoljenje kontrole letenja. Ko se z letalom nahaja na vzletni stezi, naj bi to pomenilo, da je izpolnil vse podnaloge glavne naloge talne procedure in lahko prične izvajati naslednjo nalogo ob predpostavki, da je pridobil tudi dovoljenje za izvedbo vzleta.

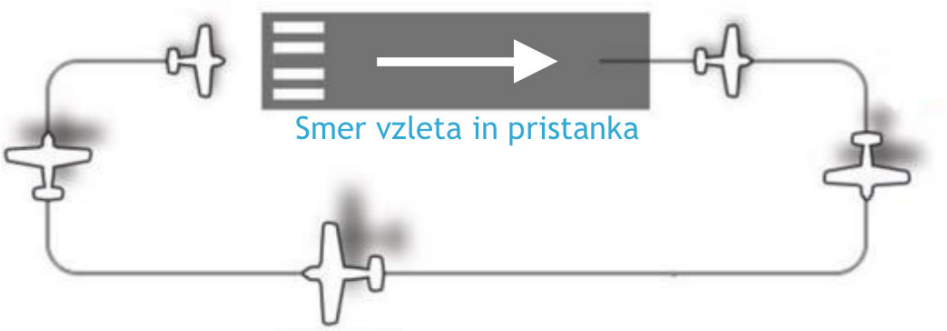
### 2.2.3 Vzlet in vzpenjanje

Sem spadajo podnaloge, ki jih pilot izvaja v fazi vzletanja in vzpenjanja na želeno višino. V fazi leta, ki se nahaja na območju letališča, se mora pilot držati točno predpisanega postopka letenja, ki v splošnem velja za vsa letališča. Na sliki 2.2 je prikazan šolski krog[4] (angl. *traffic pattern*), ki prikazuje smernice, ki jih morajo piloti upoštevati, ko se nahajajo na območju letališča. Te smernice se upoštevajo tako pri letenju zgolj v letališkem zračnem prostoru, kot tudi pri izstopanju ali vstopanju vanj. Smer (leva ali desna) šolskega kroga nikoli ni fiksno določena, temveč jo določi kontrola letenja, ki se prilagaja trenutni situaciji v zračnem prostoru. Pilot torej izvaja podnaloge ob tem, da

[3] To so vnaprej določeni seznami nalog, ki jih mora pilot po pravilih v različnih fazah leta opraviti, da lahko nadaljuje z izvajanjem poleta

[4] Narisan je desni šolski krog, lahko bi bil tudi levi, ob tem bi se krog preslikal čez os vzletne steze

upošteva ta vnaprej predpisan postopek, prilagojen trenutnim navodilom kontrole. Pilot spet izvede za to fazo leta predpisane *checkliste* in upravlja letalo tako, da sledi načrtu leta, ki ga je pripravil. Ob zapustitvi zračnega prostora letališča letalo tudi zamenja frekvenco, na kateri poteka komunikacija z zračno kontrolo in se poveže s tisto, ki je pristojna za dani zračni prostor, v katerem se letalo nahaja ali vanj vstopa.




**Slika 2.2** Simbolična slika standardnega desnega šolskega kroga (angl. *traffic pattern*) [4].

### 2.2.4 Letenje

Nalogo letenja lahko v grobem razdelimo na dva "načina" izvajanja letalskih operacij, ki sta sledeča:

- izvajanje šolskih krogov,
- letenje od neke poljubne točke A do neke poljubne točke B.

#### Izvajanje šolskih krogov

Izvajanje šolskih krogov je koristno za trening pristajanja in vzletanja, saj se tako piloti izvajanja teh faz leta tudi naučijo. Šolski krog poteka po sledečih korakih. Naše letalo vzleti, doseže določeno višino in zavije (po sliki 2.2) desno. Pri tem se še vedno vzpenja in ob določenih pogojih zopet zavije desno, vzpenja pa se, dokler ne doseže predpisane višine 1000ft [5] nad stezo letališča. Potem letalo nadaljuje let vzporedno s stezo na predpisani višini in zopet ob določenih pogojih dvakrat zavije desno, pri čemer se sedaj kontrolirano spušča, dokler ni poravnano s stezo in pristane. Podnaloge, ki jih pilot pri tem izvaja,

[5] V letalstvu se kot merska enota za izražanje višine uporabljajo čevlji (angl. *feet*), 1000ft je torej 304.8m

so definirane na sliki 2.2 v angleškem jeziku. Te v večini vsebujejo izvedbo različnih *checklist*, izvzete pa so druge pomembne podnaloge, ki smo jih spoznali že pri prejšnjih primerih. Potrebna je torej komunikacija z zračno kontrolo in prilagajanje navodilom, ki jih kontrola da pilotu. Pri izvajanju te letalske operacije lahko navigiranje zanemarimo, saj se nahajamo na območju letališča, kjer nam vzletno-pristajalna steza vseskozi ostaja v vidnem polju.

Letenje od neke poljubne točke A do neke poljubne točke B

Izvajanje te letalske operacije vključuje podnaloge izvajanja za to fazo leta predpisanih *checklist*, komuniciranja z zračno kontrolo in upoštevanja njihovih navodil ter upravljanje letala na tak način, da se bo izvedba leta čimbolj držala načrta poleta. Pri tej nalogi je nova podnaloga navigiranje. Pilot mora v vsakem trenutku leta vedeti točno kje se nahaja, ter kdaj lahko pričakuje prihod na naslednjo točko, definirano v načrtu poleta. Ob tem si lahko pomaga z različnimi navigacijskimi sredstvi, ki so na voljo v danem letalu. Ko se letalo približa ciljnemu letališču, lahko pilot začne z izvajanjem naslednje naloge na časovnem traku.

### 2.2.5 Spust in približevanje

Cilj naloge "spust in približevanje" (angl. *descent and approach*) je, da pilot letalo pripravi na pristanek na ciljnem letališču. Podnalogi, ki ju je potrebno opraviti za uspešno izvedbo te naloge, sta vzpostavitev komunikacije s kontrolnim stolpom ciljnega letališča, ter sklenitev dogovora, kako bo potekal preostanek leta, dokler se ne bo začel izvajati pristanek. Ob sledenju temu dogovoru pilot izvaja *checkliste*, predpisane za to fazo leta. Ko je letalo dovolj blizu letališča, lahko pilot začne postopek pristajanja na ciljnem letališču.

### 2.2.6 Pristanek

Cilj te naloge je, da pilot letalo spravi iz zraka na tla v mirujoče stanje. Pilot za izvedbo te naloge potrebuje eksplicitno dovoljenje zračne kontrole. Prav tako pilot ob pristanku ali pred njim dobi navodila za vožnjo do parkirnega mesta na letališču. Pilot mora torej ob upoštevanju navodil zračne kontrole upravljati letalo tako, da bo uspešno pristalo na pristajalni stezi ciljnega letališča. Ob tem se mora prilagajati trenutnim vremenskim razmeram na letališču, pri čemer je zelo pomemben dejavnik veter, o katerem mu informacije

tik pred pristankom posreduje kontrola letenja.

## 2.3 Analiza zanesljivosti socio - tehničnih sistemov

V razdelkih 2.1 in 2.2 smo v grobem opisali potek dela v opazovanem socio - tehničnem sistemu. V nadaljevanju iščemo metodo, ki nam omogoča, da zanesljivost tovrstnega sistema podrobno analiziramo. Takšna metoda mora torej ustrezno zajeti razsežnosti kompleksnih socio - tehničnih sistemov, kar pomeni, da mora upoštevati tako tehnične kot tudi človeške faktorje. Cilj analize zanesljivosti sistema je torej poiskati potencialne vzroke za odpoved sistema, da jih lahko eliminiramo še preden se nesreča zgodi [5].

### 2.3.1 Pogled na človeške napake

Upoštevanje človeškega faktorja pri analizi sistemov je lahko težavno. V sledečem odstavku skušamo na podlagi vira [6] predstaviti vpliv človeka (upravljalca sistema) na nek kompleksen socio - tehnični sistem.

Človeške napake ali potencialne človeške napake lahko vidimo na dva načina, ali kot vzrok za odpoved nekega sistema ali pa kot simptom globljih težav v sistemu. Prvi pogled na človeško napako predvideva, da bi nek socio - tehnični sistem funkcioniral normalno, če ne bi z njim upravljali nezanesljivi ljudje. Prav tako se smatra, da je človeški faktor oziroma človeška napaka vzrok za večino neželjenih dogodkov in da s tem lahko pojasnimo odpovedi sistemov. Tak pogled tudi predvideva, da človeška napaka ni del sistema, temveč je za dani sistem zgolj neprijetno presenečenje. Za nas bolj zanimiv pogled, na katerem močno sloni tudi metoda, opisana v poglavju 3, je pogled na človeške napake kot na simptome globljih težav v sistemu. Ta pogled predvideva, da sistemi sploh niso varni oziroma zanesljivi, temveč jim to zanesljivost omogoča ravno človek, torej upravljalec sistema. Na različne sisteme se namreč vršijo različne vrste pritiski, ki jih občutijo upravljalci teh sistemov, ti pritiski pa lahko ogrozijo zanesljivost oziroma varnost samih sistemov. Ljudje so edini, ki lahko sklepajo kompromise med varnostjo in temi pritiski v realnem času pod določenimi operativnimi pogoji. Človeška napaka tukaj torej ni zgolj neprijetno presenečenje, temveč posledica sposobnosti človeka, da te kompromise sklepa v spremenljivih pogojih. Če na človeško napako gledamo na ta način, torej kot del sistema, nam to omogoči, da se o njem naučimo nekaj novega. Ne iščemo torej vzroka, zaradi katerega se je upravljalec sistema zmotil, ampak se sprašujemo, zakaj se je upravljalcu sporna odločitev zdela pravilna.

V duhu tega novega pogleda na človeške napake torej iščemo neko metodo, ki primerno analizira opazovani socio - tehnični sistem.

### 2.3.2 Pregled metod in modelov

Tradicionalni pristopi k oceni zanesljivosti modernih socio - tehničnih sistemov zaradi kompleksnosti le-teh niso primerni [5,7]. V pričujočem razdelku naštejemo in opišemo nekaj izmed teh tradicionalnih pristopov, razložimo zakaj niso primerni za analizo tovrstnih sistemov ter naštejemo pristope, ki so še dokaj novi in se še uveljavljajo, vendar so primernejši za analizo modernih kompleksnih socio - tehničnih sistemov.

Nekateri izmed najbolj prakticiranih tradicionalnih pristopov k zanesljivostni analizi sistemov so sledeči:

- FTA analiza (Fault Tree Analysis) [5,7,8],
- FMEA analiza (Failure Modes and Effects Analysis) [5,7],
- FMECA analiza (Failure Modes and Effects Criticallity Analysis) [8],
- HAZOP analiza (Hazard and operability study) [5],
- ETA analiza (Event Tree Analysis) [5].

V pričujočem delu se ne poglabljamo v podrobnosti naštetih tradicionalnih pristopov, saj ti ne zajemajo kompleksnih človeških kognitivnih napak, socialnih, organizacijskih in vodstvenih faktorjev ter kompleksnih interakcij med komponentami socio - tehničnega sistema [5,8]. To je torej skupni imenovalec tem pristopom, ki so si med sabo sicer različni. Potrebujemo torej nek pristop, ki nam bo pri analizi kompleksnih socio - tehničnih sistemov omogočil zajetje vseh faktorjev, ki jih tradicionalni pristopi zanemarijo.

Na tem področju sta se v zadnjem času pojavili predvsem dve metodi - STPA (*System Theoretic Process Analysis*) in FRAM (*Functional Resonance Analysis Method*). Slednja je podrobno opisana v poglavju 3. Metoda STPA bazira na modelu STAMP (*Systems Theoretic Accident Model and Processes*). Tako metodo kot tudi model je razvila Nancy Leveson z univerze MIT. STAMP model vključuje tudi faktorje socio - tehničnih sistemov, ki jih starejši pristopi k zanesljivostni analizi ne upoštevajo [5]. STPA metoda pri zanesljivostni analizi predpostavi, da je vsak neljub dogodek oziroma nesreča posledica nezadostne kontrole v sistemu [5]. Metoda se torej osredotoča na kontroliranje sistema

namesto na posamezne komponente sistema, kot je to praksa pri tradicionalnih pristopih k zanesljivostnim analizam sistemov [9]. Namesto diagramov komponent sistema, ki jih uporabljajo tradicionalne metode zanesljivostnih analiz, STPA uporablja funkcijske kontrolne diagrame [5], ki ponazarjajo tako komponente sistema (tehnične in človeške), kot tudi tako imenovane kontrolne akcije (angl. *control action*), ki se v sistemu izvajajo (nadziranje sistema). Za vsako kontrolno akcijo se v procesu STPA analize oceni potencialno tveganje, ki ga posamezna akcija predstavlja [9]. Pri izvajanju STPA analize je treba izvesti dva glavna koraka [5]:

- identificirati potencialno nezadostno kontrolo nad sistemom, ki bi lahko vodila do tveganega stanja (angl. *hazardous state*),
- določiti na kakšen način lahko pride do nezadostnih kontrol nad sistemom, identificiranih s prvim korakom.

V viru [9] je STPA metoda aplicirana na sistem HTV (*H-II Transfer Vehicle*). HTV je brezpilotno transportno vozilo japonske vesoljske agencije (angl. textitJapan Aerospace Exploration Agency), namenjeno prevozu dobrin na mednarodno vesoljsko postajo. Po izvedbi STPA analize so avtorji članka [9] rezultate primerjali z analizo FTA (*Fault Tree Analysis*), torej z eno izmed tradicionalnih zanesljivostnih metod in ugotovili, da je STPA metoda identificirala vsa tveganja, ki jih je našla FTA, prepoznala pa je tudi faktorje, ki jih FTA metoda ni zaznala.

### 2.3.3 Povzetek problematike

V grobem smo opisali, kaj vse je potrebno za uspešno upravljanje manjšega letala, našteli smo naloge, ki se opravijo pri nekem običajnem poletu, ter jih podrobno opisali. Ugotovimo, da imamo opravka z zelo kompleksnim socio - tehničnim sistemom, v čigar analizo bi lahko vključili še vse naloge, ki so potrebne za pravilno delovanje tega socio - tehničnega sistema. To so naloge, ki jih izvajajo kontrolorji letenja, letališki delavci, serviserji letal ipd., ki jih v tem poglavju nismo naštevali. Opisali smo težave, ki se pojavijo pri zanesljivostni analizi takšnega kompleksnega socio - tehničnega sistema in pojasnili, zakaj tradicionalni pristopi k analizi tovrstnih sistemov niso zadostni. Identificirali smo dve metodi, ki sta primernejši za analizo zanesljivosti teh sistemov v primerjavi s klasičnimi zanesljivostnimi metodami. V naslednjih poglavjih za izvedbo analize zanesljivosti vzorčnega socio - tehničnega sistema uporabimo FRAM metodo, ki je opisana v

poglavju 3. Ugotovimo, da je sistem, ki smo ga opisali v razdelku 2.2, prevelik, da bi ga v celoti analizirali v pričujočem delu, vendar izjemno primeren za analizo s FRAM metodo, saj je tipičen primer socio - tehničnega sistema. V poglavju 4 se tako osredotočimo zgolj na en del opazovanega sistema. Ta del je označen na sliki 2.1 in se nahaja med točkama "Točka 1" in "Točka 2". Vsebuje torej nekatere elemente naloge talne procedure, ter elemente izvajanja vzleta. Ta del je torej zelo pomemben del izvajanja letalskih operacij.

# 3 Opis FRAM metode

Problem in potrebo po FRAM metodi smo pojasnili že v poglavju 2. Metoda FRAM je bila razvita kot odgovor na ta problem. Metodo je razvil profesor Erik Hollnagel z danske univerze Syddansk Universitet. V pričujočem poglavju to metodo podrobneje opišemo na podlagi virov [10] in [2].

## 3.1 Principi FRAM metode

Metoda FRAM se v prvih analitičnih korakih osredotoča na socio - tehnični sistem v njegovem "normalnem" stanju. V začetku nas torej zanima normalno delovanje sistema, ne pa odpovedi le-tega. Metodo lahko uporabimo tako za zanesljivostno analizo nekega socio - tehničnega sistema, kot tudi za analizo odpovedi tovrstnega sistema. Uporabimo jo lahko torej za morebitne bodoče dogodke, kot tudi za pretekle dogodke. Cilj metode je izgraditi model, ki opisuje kako stvari potekajo, namesto interpretirati dogajanje iz nekega modela. Metoda sloni na štirih osnovnih principih, opisanih v pričujočem razdelku.

### 3.1.1 Princip ekvivalence uspeha in odpovedi

Princip ekvivalence uspeha in odpovedi pomeni, da imata tako uspešno delovanje sistema, kot tudi odpoved sistema isti izvor. Uspeh in neuspeh se torej zgodita iz istih razlogov. Zakaj je temu tako, pojasni princip približnih prilagoditev, pojasnjen v razdelku 3.1.2.

### 3.1.2 Princip približnih prilagoditev

Tehnične komponente v socio - tehničnih sistemih so narejene tako, da delujejo konstantno na isti način. Za upravljalce teh sistemov to ne velja, saj je človeško delovanje vedno variabilno iz različnih razlogov. V socio - tehničnih sistemih pogoji za delo mnogokrat niso enaki tistim, ki so predpisani ali predvideni. Za uspešno upravljanje sistema se je torej potrebno neprestano prilagajati trenutnim delovnim razmeram, ker pa so sredstva[1], potrebna za delovanje nekega sistema mnogokrat omejena, so te prilagoditve bolj približne kot natančne. Kjub temu so večinoma dovolj dobre za izvedbo zadane naloge in pojasnijo, zakaj sistem uspešno deluje na vsakodnevni bazi. Zaradi teh približnih prilagoditev pa gredo stvari včasih tudi narobe, čeprav v veliki večini primerov sistem zaradi njih deluje pravilno.

Ta princip, združen s principom ekvivalence, ki je opisan v razdelku 3.1.1, močno spominja na nov pogled na človeške napake, predstavljen v razdelku 2.3.1. Socio - tehnični sistemi torej delujejo, ker se upravljalci le-teh prilagodijo delovnim razmeram, včasih pa zaradi teh prilagoditev tudi odpovejo.

### 3.1.3 Princip pojavitve

Kadar iščemo razlago za nek nepričakovan dogodek v sistemu, se mnogokrat opremo na klasične razlage, ki uporabljajo princip kavzalnosti[2] in dekompozicije. Te razlage torej predpostavijo, da je nek nepričakovan pojav **rezultat** (ne)delovanja znanih komponent in procesov sistema.

Včasih pa nepričakovan dogodek ne more biti pojasnjen zgolj kot rezultat znanih procesov sistema. V teh primerih lahko rečemo, da je nepričakovan dogodek **pojav** in ne **rezultat**. Še vedno je mogoče pojasniti kaj se je zgodilo, vendar ne s principoma dekompozicije in kavzalnosti. Vzroki za nepričakovan dogodek so namreč lahko **izmuzljivi**

[1] Npr. material, informacije, čas, ...

[2] Princip kavzalnosti je definiran kot razmerje med vzrokom in posledico in predvideva, da ima vse svoj vzrok

(angl. elusive). To pomeni, da teh vzrokov nikoli ne moremo "najti"[3], temveč jih lahko zgolj rekonstruiramo. To so vzroki oziroma okoliščine, ki so obstajale zgolj omejen čas (so minljive). To pomeni, da teh vzrokov za odpoved ne moremo direktno odstraniti ali popraviti, lahko pa morda nadziramo okoliščine, ki so do pojavitve (angl. *emergence*) teh vzrokov pripeljale.

### 3.1.4 Princip resonance

V fiziki je resonanca pojav, ko zunanja sila deluje na neko telo tako, da to začne nihati z večjo amplitudo. Podoben pojav se lahko pojavi tudi pri sistemskih funkcijah socio - tehničnih sistemov. Metoda FRAM takšen pojav imenuje **funkcijska resonanca**. Princip približnih prilagoditev, pojasnjen v razdelku 3.1.2 nam pove, da je človeško delovanje variabilno. V večini primerov je ta variabilnost tako majhna, da za opazovani sistem nima neželenih posledic. Ko pa se naenkrat zgodi več približnih prilagoditev v sistemu, se variabilnost sistema lahko opazno poveča. Temu pojavu rečemo **funkcijska resonanca**. Gre torej za opazen izid ali signal, ki se **pojavi** zaradi nenamerne interakcije med različnimi variabilnimi signali, ki so variabilni zaradi vsakodnevnih približnih prilagoditev. Ta variabilnost pa ni čisto naključna, saj se približne prilagoditve namreč izvajajo po nekih vedenjskih vzorcih, ki so do neke mere predvidljivi.

## 3.2 Funkcije in aspekti

V FRAM metodi pri modeliranju opazovanega socio - tehničnega sistema funkcija predstavlja osnovni gradnik. Funkcija predstavlja neko aktivnost, ki jo je treba izvesti, da se doseže nek rezultat. Funkcije klasificiramo na dva načina, in sicer glede na vlogo, ki jo v FRAM modelu funkcija ima ter glede na to, kdo ali kaj funkcijo izvaja. Glede na pomembnost jih lahko razdelimo na **funkcije ozadja** ter **funkcije ospredja**.

Funkcija spada v kategorijo funkcij ospredja, če smatramo, da ima njena variabilnost lahko vpliv na nek dogodek, ki ga preiskujemo. Za funkcijo ozadja lahko torej sklepamo, da ni variabilna.

Funkcije lahko razdelimo še glede na to, kdo jo izvaja. Funkcije so torej lahko:

- človeške, če jih izvaja posameznik ali skupina ljudi,
- organizacijske, če jih izvaja organizacija kot celota,

[3]Nekatere vzroke, npr. fizične komponente nekega sistema lahko po odpovedi najdemo in ocenimo

- tehnološke, če jih izvaja tehnični del sistema.

FRAM metoda predvideva, da imajo tehnološke funkcije zanemarljivo variabilnost, saj se predpostavlja, da so tehnične komponente stabilne. Pri organizacijskih funkcijah se način delovanja spreminja počasi (frekvenca), vendar so razlike med rezultati (amplituda) lahko ogromne. Človeške funkcije imajo po naravi tako visoko frekvenco, kot tudi amplitudo. V praksi to pomeni, da se delovanje človeških funkcij lahko spreminja zelo hitro (frekvenca), tako na boljše kot na slabše, razlike v delovanju pa so lahko ogromne (amplituda).

Splošno pravilo je, da se za naziv funkcije uporabi glagol ali glagolsko besedno zvezo. Funkcije lahko identificiramo s pomočjo opisa poteka dela v opazovanem socio - tehničnem sistemu.

### 3.2.1 Aspekti funkcij

V okviru FRAM metode funkcijo definiramo s pomočjo šestih aspektov. Ni potrebno, da je za neko poljubno funkcijo definiranih vseh šest, vendar so lahko definirani zgolj tisti, ki se zdijo analitiku sistema potrebni. Funkcija ima lahko več instanc enega aspekta. Vse funkcije ospredja morajo imeti definiran vsaj en vhod in en izhod. Splošno pravilo je, da so nazivi aspektov samostalniki ali samostalniške besedne zveze, saj so aspekti stanja ali rezultati nečesa in ne aktivnosti. Aspekti, ki jih FRAM metoda predpisuje za opredelitev funkcije, ki je vizualizirana na sliki 3.1, so sledeči:

- input/vhod: Vhod je definiran kot tisto, kar običajno funkcija pretvori v izhod. V praksi je lahko to naprimer material, informacija in podobno. Pri FRAM metodi je lahko vhod tudi tisto, kar zažene funkcijo, torej začetni pogoj za izvajanje funkcije.

- output/izhod: Izhod iz funkcije je rezultat tistega, kar funkcija izvede. Tako kot pri vhodu, je to lahko naprimer informacija, signal za začetek neke druge funkcije, itd.

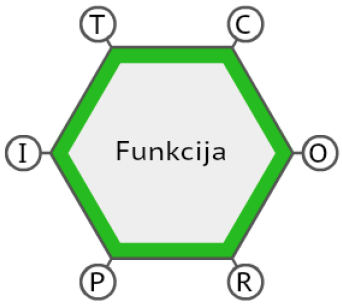
- precondition/pogoj: Pogoje razumemo kot stanja, ki morajo biti izpolnjena, preden se lahko funkcija izvede. Pogoja ne smemo razumeti kot signal za začetek izvajanja neke funkcije; to funkcijo vrši aspekt vhod. Pogoj naj bi vedno bil izhod iz neke druge funkcije.

- resource/vir: Vir je definiran kot tisto, kar se med izvajanjem funkcije porablja. To je torej snov, energija, moč, informacija in podobno. Tudi čas bi lahko spadal pod

ta aspekt vendar ima pri FRAM analizi poseben status in predstavlja samostojen aspekt.

- control/nadzor: Nadzor je definiran kot tisto, kar nadzira izvajanje neke funkcije. Pod ta aspekt se lahko šteje nek načrt, urnik, procedure, sklop navodil ali predpisov, ki jim mora funkcija slediti in podobno. Nadzor je lahko tudi socialnega značaja, naprimer pričakovanja sodelavcev ali organizacije ali pričakovanja do samega sebe.

- time/čas: Čas lahko razumemo na več načinov; lahko naprimer definiramo zaporedje, po katerem se morajo določene funkcije izvesti, morda pa se morajo izvesti celo istočasno. Ta aspekt lahko definiramo tudi kot trajanje funkcije.




**Slika 3.1** Vizualizacija funkcije FRAM modela, ki je običajno prikazana s šestkotniki, vsak kot ima svoj aspekt: I - input, O - Output, P - Precondition, R - Resource, C - Control, T - Time (narejeno s programskim orodjem FRAM Model Visualiser [11]).

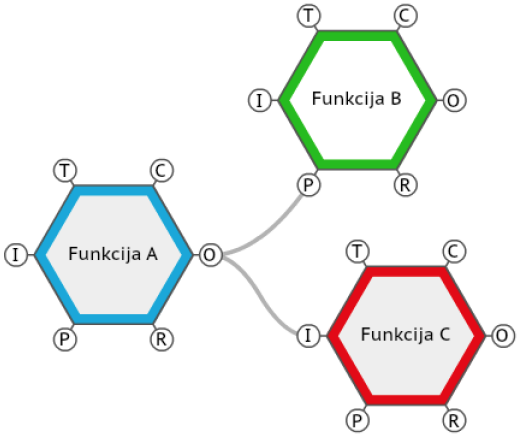
### 3.2.2 Odnosi med funkcijami

Funkcije v FRAM modelu so torej definirane z aspekti, opisanimi v razdelku 3.2.1. Če imata dve različni funkciji isti imeni aspektov (naprimer izhod iz ene funkcije in vhod v neko drugo funkcijo), obstaja potencialna odvisnost oziroma spoj (angl. *coupling*) med tema dvema funkcijama. Izhod iz funkcije A torej funkcija B uporabi kot svoj vhod. Relacije v FRAM modelu niso tipa **1-1**, temveč so tipa **n-n**. Neka funkcija ima torej lahko več instanc enega aspekta, kot smo že povedali v začetku razdelka 3.2.1. Funkcija ima lahko torej naprimer dva različna vhoda, ki prihajata iz dveh različnih funkcij. En izhod iz neke funkcije gre lahko tudi na več različnih funkcij v različne aspekte le-teh, kot

prikazuje slika 3.2. Analiza FRAM modela torej poteka tako, da sledimo potencialnim spojem med funkcijami in se sprašujemo, katere aspekte še potrebujemo, da se nam bo obseg opisa sistema zdel zadosten. S tem tudi odkrivamo nove funkcije. Meje modela torej določi analitik.

FRAM model, ki ga razvijemo, opisuje neko **tipično** situacijo v sistemu, ne pa **specifične**. Ne moremo torej trditi, da se bo neka poljubno izbrana funkcija zgodila pred neko drugo funkcijo; to namreč lahko trdimo šele, ko naredimo **instanco modela**. To pomeni, da s podrobnimi informacijami o neki situaciji ustvarimo primer modela. Takrat pa lahko funkcije glede na zaporedje izvajanja razdelimo v dve kategoriji - predhodne (angl. *upstream*) funkcije - torej tiste, katere so se že izvedle in naslednje (angl. *downstream*) funkcije - tiste, ki se še bodo.

FRAM model lahko za lažjo predstavo vizualiziramo, vendar je osnova za analizo vedno tekstovni opis FRAM modela.




**Slika 3.2** Primer spojev med funkcijami FRAM modela.

## 3.3 Opis variabilnosti

V razdelku 3.1.4 je opisan princip funkcijske resonance, ki je posledica variabilnosti posameznih funkcij. FRAM metoda predpisuje merila, s katerimi to variabilnost formalno opišemo in kategoriziramo. Variabilnost funkcij razdelimo v dve kategoriji, **potenicalno variabilnost** za splošni model socio - tehničnega sistema, ter **dejansko variabilnost**

za neko instanco modela tega sistema. Variabilnost izvedbe opazovane funkcije nas pri analizi zanima zgolj, če je variabilen tudi **izhod** te funkcije. Če je izhod funkcije v merilih variabilnosti konstanten, je variabilnost izvedbe funkcije torej za nas nepomembna. Variabilnost izhoda funkcije se lahko pojavi iz naslednjih treh različnih razlogov:

- Variabilnost izhoda je lahko zgolj posledica variabilnosti izvedbe funkcije. Takšno variabilnost imenujemo **notranja** ali **endogena** variabilnost.
- Variabilnost izhoda se lahko pojavi zaradi variabilnosti delovnih razmer in okolja. Takšna variabilnost se imenuje **zunanja** ali **eksogena** variabilnost.
- Variabilnost izhoda se lahko pojavi zaradi vplivov **predhodnih** funkcij, katerih izhodi so variabilni. Na tem torej sloni princip funkcijske resonance.

Variabilnost glede na vrste funkcij (tehnološke, človeške in organizacijske) smo opisali na začetku razdelka 3.2. FRAM metoda se najbolj osredotoča na človeške funkcije. Potrebujemo torej orodje, s katerim opišemo, kako se ta variabilnost funkcij v modelu dejansko pojavi. To lahko storimo na dva različna načina. V pričujočem delu opišemo zgolj enostavnejši način za opisovanje variabilnosti, bolj podroben način pa je opisan v viru [2].

Enostavnejši način za opisovanje variabilnosti izhoda funkcije le to predstavi z vidika **časa** in **natančnosti**. Časovno se lahko izhod neke funkcije pojavi **prezgodaj**, **točno**, **prepozno** ali **nikoli**. Z vidika natančnosti je lahko izhod neke funkcije **natančen**, **sprejemljiv** ali **nenatančen**.

## 3.4 Koraki FRAM metode

FRAM metoda se po načelih, opisanih v razdelkih 3.1 - 3.3 izvede v štirih glavnih korakih:

1. Identifikacija in opis funkcij: Potrebno je najti funkcije sistema in jih karakterizirati s pomočjo šestih aspektov, razvitih za ta namen.
2. Identifikacija variabilnosti: Potrebno je identificirati potencialno variabilnost FRAM modela, ter dejansko variabilnost za eno ali več instanc tega modela.
3. Agregacija variabilnosti - funkcijska resonanca: Glede na spoje med funkcijami in njihovimi variabilnostmi je potrebno določiti morebitno funkcijsko resonanco.

4. Razvoj priporočil za preprečitev funkcijske resonance: Glede na ugotovitve iz prvih treh korakov poizkušamo razviti priporočila za omejevanje variabilnosti določenih funkcij, da bi preprečili neželjeno funkcijsko resonanco.

Na podlagi znanja, pridobljenega v pričujočem poglavju, lahko sedaj FRAM metodo apliciramo na specifičen primer.

# 4 Izvedba FRAM analize vzorčnega primera

Za problematiko, predstavljeno v poglavju 2, v pričujočem poglavju po vodilih FRAM metode oblikujemo model, ki karseda natančno prikazuje, kako se naloge v danem delovnem okolju opravljajo in kako so med seboj povezane. Osredotočimo se na del poleta, ko letalo izvaja "šolski krog", predstavljen na sliki 2.2. Predpostavimo, da se nahajamo na vstopni točki pred vzletno stezo, kjer stojimo pri miru in se pripravljamo na vzlet. Naloge, ki se morajo v naslednjih korakih v delu našega socio - tehničnega sistema izvesti, zagotavljajo uspešen vzlet letala. V naslednjem razdelku tako realiziramo prepoznavo in opis pomembnih sistemskih funkcij ter njihovo karakterizacijo, ki se določi s pomočjo šestih osnovnih karakteristik oziroma aspektov. V razdelku 4.2 je model, ki je opisan v razdelku 4.1 vizualiziran, v razdelku 4.3 je analiziran vzorčni primer oziroma instanca funkcijske resonance tega modela, v razdelku 4.4 pa se osredotočimo na deklaracijo protokola pri funkcijski resonanci.

## 4.1 Identifikacija in karakterizacija pomembnih sistemskih funkcij

V pričujočem razdelku naštejemo in na kratko opišemo najpomembnejše funkcije opazovanega socio - tehničnega sistema. Omenjene funkcije so sledeče:

- izvajanje funkcije "ground check checklist",
- izvajanje funkcije "before takeoff checklist",
- sprejemanje navodil zračne kontrole,
- interpretacija navodil zračne kontrole,
- pridobitev dovoljenja za vzlet,
- izvajanje funkcije "readback",
- ATC preverjanje,
- prilagajanje vzletnim navodilom kontrole,
- izvajanje vzleta.

### 4.1.1 Izvajanje funkcije "ground check checklist"

Letalo pred izvedbo te naloge s prižganim motorjem miruje pred vzletno stezo. Namen te funkcije je, da pilot preveri, ali je letalo pripravljeno za polet. Tekom izvajanja te funkcije se preveri, če je motor dovolj ogret, če delujejo vžigalni magneti, če deluje gretje vplinjača in alternator. Funkcija ima izreden pomen za nadaljnji potek leta; v primeru, da se opazi napaka pri delovanju katerekoli izmed komponent sistema, se let lahko prekine, če pa napaka obstaja in ostane neopažena, je rezultat lahko katastrofalen. Letalo po izvedeni funkciji še vedno stoji pri miru. Ta funkcija spada v kategorijo **človeških funkcij**, ter v kategorijo **funkcij ospredja**. Podrobna analiza funkcije je prikazana v tabeli 4.1.

### 4.1.2 Izvajanje funkcije "before takeoff checklist"

Po uspešni izvedbi funkcije "ground check checklist" se letalo še vedno nahaja na isti točki kot prej. Funkcija je namenjena še zadnjim pripravam na polet. Pilot nastavi

| Aspekt | Opis aspekta |
|---|---|
| Input/Vhod | Letalo se nahaja pred vzletno stezo |
| Output/Izhod | Sistem je pripravljen za nadaljevanje postopka |
| Precondition/Pogoj | / |
| Resource/Vir | Čas |
| Control/Nadzor | Ground check checklist |
| Time/Čas | -Lahko obstaja časovni pritisk zaradi povečanega prometa<br>-Nujno opraviti pred ostalimi funkcijami |

**Tabela 4.1** Analiza funkcije [Izvajanje "ground check checklist"].

večino nastavitev v letalu tako, da je letalo nastavljeno v konfiguraciji za vzlet, preden zaprosi za dovoljenje za vzlet, oziroma sporoči, da je na vzlet pripravljen. Po izvedeni funkciji letalo še vedno miruje. Ta funkcija spada v kategorijo **človeških funkcij**, ter v kategorijo **funkcij ospredja**. Podrobna analiza funkcije je prikazana v tabeli 4.2.

| Aspekt | Opis aspekta |
|---|---|
| Input/Vhod | Sistem je pripravljen za nadaljevanje postopka (izhod funkcije [Izvajanje "ground check checklist"]) |
| Output/Izhod | Lahko zaprosimo za dovoljenje za vzlet |
| Precondition/Pogoj | / |
| Resource/Vir | Čas |
| Control/Nadzor | Before takeoff checklist |
| Time/Čas | -Lahko obstaja časovni pritisk zaradi povečanega prometa<br>-Nujno opraviti pred ostalimi funkcijami |

**Tabela 4.2** Analiza funkcije [Izvajanje "before takeoff checklist"].

### 4.1.3 Sprejemanje navodil zračne kontrole

Ta funkcija ponazarja vsak sprejem navodil zračne kontrole in je namenjena zgolj lažji predstavi modela. To funkcijo kategoriziramo kot **človeško funkcijo**, ter kot **funkcijo ozadja**, kar pomeni da njeno obnašanje oziroma njeni rezultati ne varirajo preveč. Kontrolor poda navodila, kakršnakoli pač so, pilot pa jih preko radio zveze sprejme. Ali je

pilot navodila sprejel in interpretiral pravilno, se preverja v funkciji [Izvajanje funkcije "readback"]. Podrobna analiza funkcije je prikazana v tabeli 4.3.

| Aspekt | Opis aspekta |
|---|---|
| Input/Vhod | / |
| Output/Izhod | Navodila zračne kontrole |
| Precondition/Pogoj | / |
| Resource/Vir | / |
| Control/Nadzor | / |
| Time/Čas | / |

**Tabela 4.3** Analiza funkcije [Sprejemanje navodil zračne kontrole].

### 4.1.4 Interpretacija navodil zračne kontrole

Pilot sprejme navodila zračne kontrole preko radio zveze in jih interpretira. Pri tem uporabi kompetence, pridobljene pri šolanju in praktičnih izkušnjah ter pravila letalske frazeologije. Mnogokrat se lahko pilot pri miselnem procesu interpretacije zmoti in s tem napačno razume podana navodila. Ta funkcija spada v kategorijo **človeških funkcij**, ter v kategorijo **funkcij ospredja**. Podrobna analiza funkcije je prikazana v tabeli 4.4.

| Aspekt | Opis aspekta |
|---|---|
| Input/Vhod | Navodila zračne kontrole (izhod funkcije [Sprejemanje navodil zračne kontrole]) |
| Output/Izhod | Interpretirana navodila zračne kontrole |
| Precondition/Pogoj | / |
| Resource/Vir | Kompetence pilota |
| Control/Nadzor | Pravila letalske frazeologije |
| Time/Čas | / |

**Tabela 4.4** Analiza funkcije [Interpretacija navodil zračne kontrole].

### 4.1.5 Pridobitev dovoljenja za vzlet

Pilot ob pogoju, da je uspešno izvedel funkciji [Izvajanje "ground check checklist"] ter [Izvajanje "before takeoff checklist"] letališko kontrolo letenja zaprosi za dovoljenje za vzlet oziroma sporoči, da je na vzlet pripravljen, v kolikor dovoljenja še nima. Kontrola mu lahko v danem trenutku vzlet odobri ali pa sporoči, da naj na dovoljenje počaka. Letalo mora ob zavrnitvi ostati tam kjer je, razen če dobi drugačna navodila letališke kontrole. Možno je tudi, da letališka kontrola iz različnih razlogov odobri zgolj premik na vzletno stezo, samega vzleta pa v danem trenutku še ne. V tem primeru se letalo lahko premakne na vzletno stezo, vendar mora počakati na dovoljenje za vzlet. Ta funkcija spada v kategorijo **človeških funkcij**, ter v kategorijo **funkcij ospredja**. Podrobna analiza funkcije je prikazana v tabeli 4.5.

| Aspekt | Opis aspekta |
|---|---|
| Input/Vhod | Interpretirana navodila zračne kontrole (izhod funkcije [Interpretacija navodil zračne kontrole]) |
| Output/Izhod | Letalo lahko vzleti |
| Precondition/Pogoj | Lahko zaprosimo za dovoljenje za vzlet (izhod funkcije [Izvajanje "before takeoff checklist"]) |
| Resource/Vir | / |
| Control/Nadzor | Popravek ali potrditev (izhod funkcije [ATC preverjanje]) |
| Time/Čas | / |

**Tabela 4.5** Analiza funkcije [Pridobitev dovoljenja za vzlet].

### 4.1.6 Izvajanje funkcije "readback"

Pravila letalske komunikacije določajo, da mora pilot vsakič, ko prejme navodila kontrole letenja, kontrolorju ponoviti pravkar izrečena navodila. S tem se tako kontrolor, kot tudi pilot prepričata, da ni prišlo do nikakršnega nesporazuma v komunikaciji in lahko nadaljujeta vsak svoja opravila. Ta funkcija je izrednega pomena za naš socio - tehnični sistem, saj se pojavi pri večini komunikacij s kontrolo letenja. Ta funkcija spada v kategorijo **človeških funkcij**, ter v kategorijo **funkcij ospredja**. Podrobna analiza funkcije je prikazana v tabeli 4.6.

| Aspekt | Opis aspekta |
|---|---|
| Input/Vhod | Interpretirana navodila zračne kontrole (izhod funkcije [Interpretacija navodil zračne kontrole]) |
| Output/Izhod | Ponovljena navodila kontrole |
| Precondition/Pogoj | / |
| Resource/Vir | / |
| Control/Nadzor | / |
| Time/Čas | / |

**Tabela 4.6** Analiza funkcije [Izvajanje funkcije "readback"].

### 4.1.7 ATC preverjanje

Funkcijo [ATC (angl. *Air Traffic Control*) preverjanje] izvaja pristojni kontrolor letenja. Z izvajanjem te funkcije se preveri, če je pilot dana navodila razumel pravilno. Če kontrolor v pilotovih informacijah zazna napako, ga mora na njo opomniti, lahko pa se zgodi, da se ta napaka spregleda. Ta funkcija spada v kategorijo **človeških funkcij**, ter v kategorijo **funkcij ospredja**. Podrobna analiza funkcije je prikazana v tabeli 4.7.

| Aspekt | Opis aspekta |
|---|---|
| Input/Vhod | Ponovljena navodila kontrole (izhod funkcije [Izvajanje funkcije "readback"]) |
| Output/Izhod | Popravek ali potrditev |
| Precondition/Pogoj | / |
| Resource/Vir | / |
| Control/Nadzor | / |
| Time/Čas | / |

**Tabela 4.7** Analiza funkcije [ATC preverjanje].

### 4.1.8 Prilagajanje vzletnim navodilom kontrole

Pilot ob dovoljenju za vzlet, lahko pa tudi preje prejme tudi navodila, katero smer steze naj uporabi. Kot je razvidno iz slike 4.1, je fizično na Ljubljanskem letališču prisotna zgolj ena steza, ki pa ima dve smeri; 30, kar označuje magnetno smer 300° (v tem primeru natančneje 304°), in 12, kar označuje magnetno smer 120° (v tem primeru natančneje 124°). Če predpostavimo, da se nahajamo na točki TWY F na sliki 4.1, je z našim letalom iz tega izhodišča fizično mogoče vzleteti v obe smeri. Dogovorjeno smer mora torej pilot strogo upoštevati, saj bi v primeru napake vzletel v popolnoma nasprotno smer, kot to pričakuje kontrola letenja, kar pa ima lahko za naše letalo, kot tudi za soudeležence v letalskem prometu katastrofalne posledice. Ta funkcija spada v kategorijo **človeških funkcij**, ter v kategorijo **funkcij ospredja**. Podrobna analiza funkcije je prikazana v tabeli 4.8.

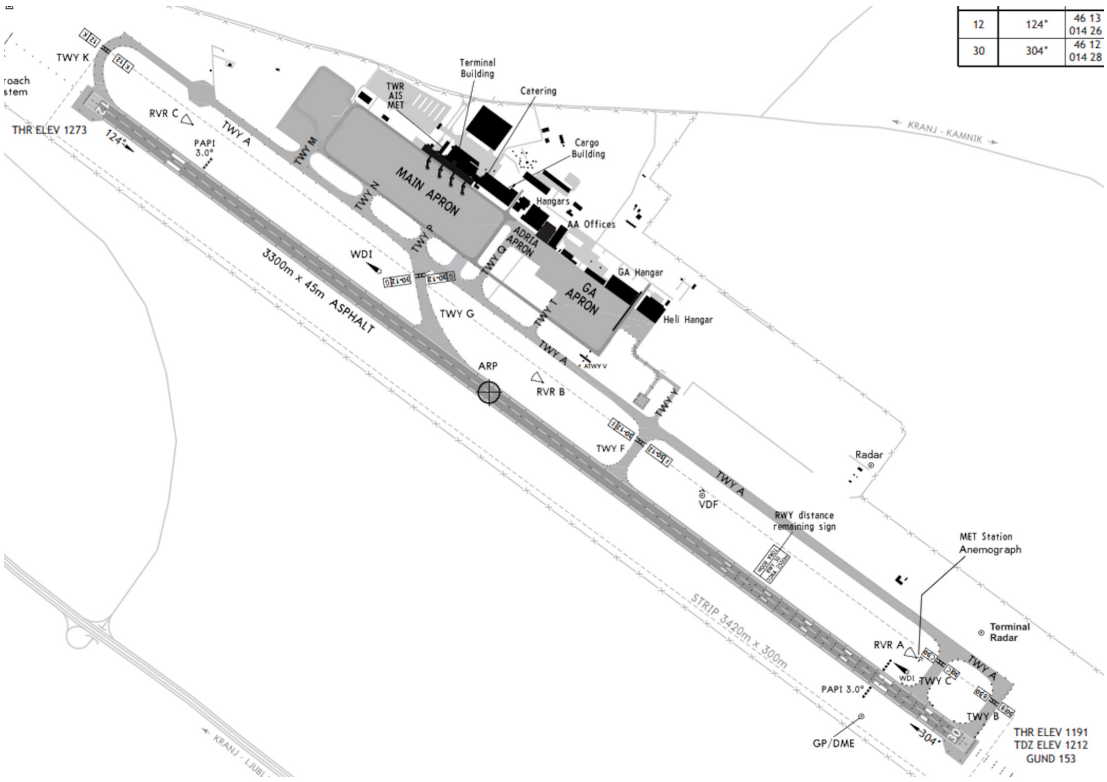
| Aspekt | Opis aspekta |
|---|---|
| Input/Vhod | Interpretirana navodila zračne kontrole (izhod funkcije [Interpretacija navodil zračne kontrole]) |
| Output/Izhod | Nahajamo se na vzletni stezi, obrnjeni v pravilno smer |
| Precondition/Pogoj | / |
| Resource/Vir | / |
| Control/Nadzor | Popravek ali potrditev (izhod funkcije [ATC preverjanje]) |
| Time/Čas | / |

**Tabela 4.8** Analiza funkcije [Prilagajanje vzletnim navodilom kontrole].

### 4.1.9 Izvajanje vzleta

Vzlet se izvede po točno predpisanem postopku, ki na primer specificira koliko plina je potrebno odpreti v fazi samega vzletanja, pri kolikšni hitrosti mora pilot preveriti motorske inštrumente, ter pri kolikšni hitrosti se izvede rotacija[1]. Ta funkcija spada v kategorijo **človeških funkcij**, ter v kategorijo **funkcij ospredja**. Podrobna analiza funkcije je prikazana v tabeli 4.9.

[1] Dvig nosu letala z vzletne steze pri postopku vzletanja




**Slika 4.1** Izsek iz operativne karte ljubljanskega letališča [12].

## 4.2 Vizualizacija modela opisanih funkcij

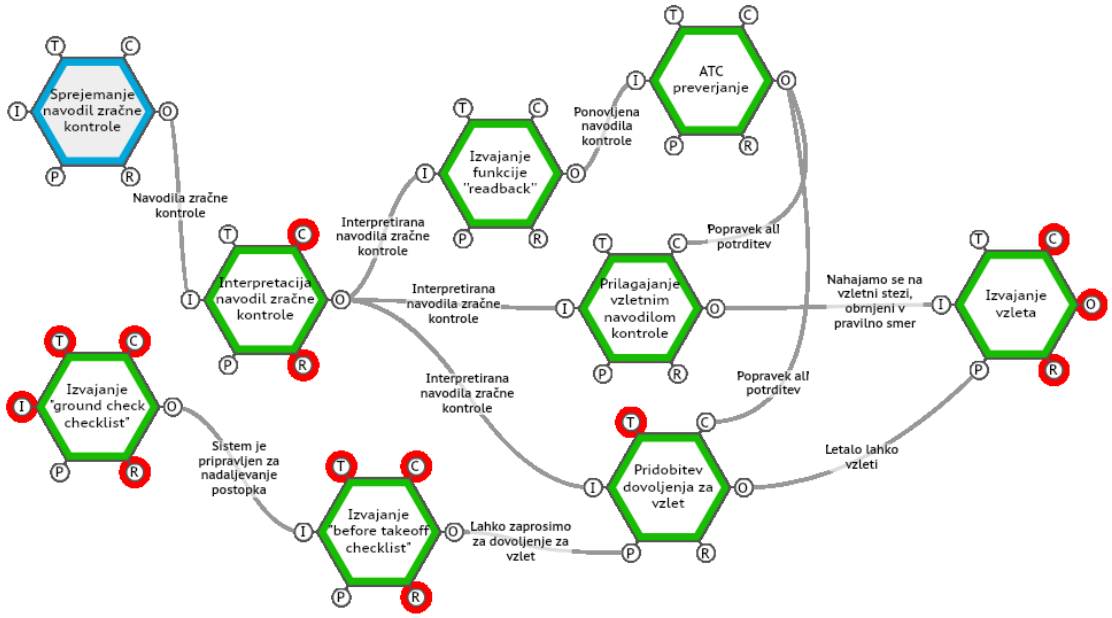
S pomočjo programskega orodja FRAM Model Visualiser [11] smo vizualizirali model, opisan v razdelku 4.1, ki si ga sedaj mnogo lažje predstavljamo. Rezultat je viden na sliki 4.2. Z zeleno so označene funkcije ospredja, z modro pa funkcije ozadja. Aspekti, označeni z rdečo barvo, so definirani aspekti.

## 4.3 Vzorčna analiza resonance

V tem razdelku se lotimo iskanja resonance v specifičnem primeru v razdelku 4.1 opisanega modela, oziroma v instanci modela. Iščemo torej primer, kjer neka funkcija na svoj izhod pošlje signal, za katerega smatra, da je v mejah normale, ki jih določijo aspekti te funkcije, do naslednje funkcije, ki ta signal sprejme, vendar je zanjo ta signal potencialno

| Aspekt | Opis aspekta |
|---|---|
| Input/Vhod | Nahajamo se na vzletni stezi, obrnjeni v pravilno smer (izhod funkcije [Prilagajanje vzletnim navodilom kontrole]) |
| Output/Izhod | Letalo je v zraku |
| Precondition/Pogoj | Sistem ima zeleno luč za vzlet |
| Resource/Vir | Gorivo |
| Control/Nadzor | Vzlet se izvede po predpisanem postopku |
| Time/Čas | / |

**Tabela 4.9** Analiza funkcije [Izvajanje vzleta].




**Slika 4.2** Vizualizacija opazovanega modela.

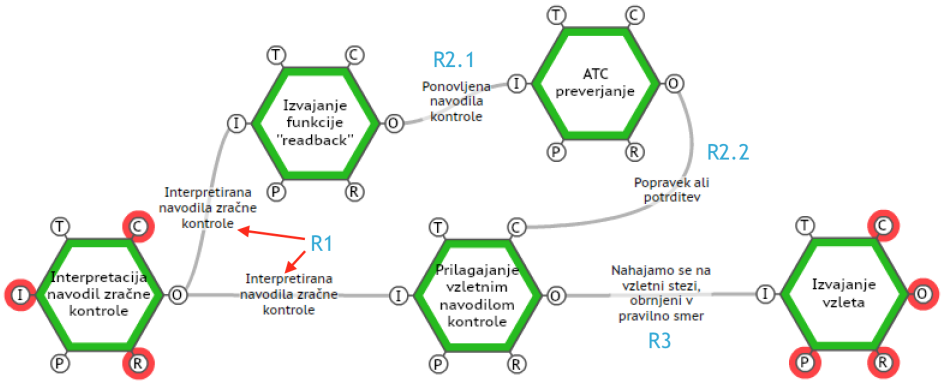
nesprejemljiv ali izven normalnih predvidenih intervalov.

Pilot se v našem modelu pripravlja na vzlet, pri tem pa naj bi pravilno izvedel vse naloge, ki so za to potrebne. Prilagodi se navodilom kontrole letenja, izvede vse potrebne *checkliste*, zapelje na stezo in vzleti v napačno smer. Tak scenarij ima lahko za naš

sistem katastrofalne posledice, v vsakem primeru pa je to neljub dogodek. Za varno izvajanje letalskih operacij je izjemno pomembna komunikacija, ki mora potekati po predpisih letalske frazeologije. Očitno se je v omenjenem scenariju nekje zgodila napaka v komunikaciji, ki sicer za vzletanje običajno izgleda tako, da letališka kontrola pilotu sporoči, da lahko vzleti, pri tem pa mu pove tudi, na kateri stezi. Pilot mora po pravilih ta navodila prebrati nazaj (izvesti funkcijo "readback"), tako da kontrolor ve, če ga je pilot razumel pravilno ali ne. Številke vzletnih stez se v letalstvu, v izogib nesporazumom, berejo kot posamezne številke; steza 13 se torej bere kot "steza ena tri" (angl. *runway one three*). Vsaka steza ima torej dve smeri. V tej instanci modela sta to torej smer 13 in smer 31. Smer steze se torej bere kot "ena tri" ali pa "tri ena", številki sta torej za obe smeri isti, zgolj v različnem vrstnem redu, kar zviša možnosti za potencialno napako pri posredovanju informacije o vzletni smeri. Takšne smeri steze so bile do nedavnega v uporabi na ljubljanskem letališču[2], trenutno pa sta smeri steze 12 in 30. Da lahko pride do omenjenega scenarija, torej da letalo zmore vzleteti v obe smeri, smo pojasnili že s primerom v razdelku 4.1.8. Pilot včasih, kar se v tem primeru po pravilih komunikacije ne bi smelo zgoditi, namesto da prebere polna navodila kontrole, uporabi frazo "wilco" ali pa "roger". S tem sporoča, da je navodila kontrole uspešno sprejel, vendar tako kontrolor ne more preveriti, če je pri komuniciranju pilot morda kaj narobe razumel. Lahko se tudi zgodi, da pilot pravilno razume navodila, vendar jih kasneje ne upošteva pravilno (pozabljanje). Ugotovimo, da **kombinacija funkcij** [Izvajanje funkcije "readback"] in [ATC preverjanje] lahko v določenih primerih, v katerih ni zanemarljiv tudi doprinos variabilnosti funkcije [Interpretacija navodil zračne kontrole], prepuščata signal, ki sporoča, da sta bili funkciji izvedeni pravilno, čeprav temu ni tako. Ta signal potem potuje do funkcije [Prilagajanje vzletnim navodilom kontrole], ki ga uporabi za aspekt **nadzor**. Če je nek aspekt izven predvidenih intervalov, kot v našem primeru je, funkcija pa se po njem ravna, to lahko vpliva na izvedbo funkcije do te mere, da je izhod funkcije drugačen od željenega. Ta izhod ima potem določen vpliv tudi na funkcijo [Izvajanje vzleta]. Na sliki 4.3 je prikazano, kako so funkcije opazovanega vzorčnega primera resonance med seboj povezane. Tabela 4.10 prikazuje, kakšna je lahko variabilnost izhodov funkcij opazovane sekcije FRAM modela. Variabilnost je opisana z upoštevanjem meril **časa** in **natančnosti**. Ugotovimo torej, da variabilnost, ki se iz

[2]Ker so smeri steze določene z magnetno smerjo, magnetno polje našega planeta pa se spreminja (magnetna pola se premikata), se posledično spremeni tudi smer steze

različnih že omenjenih razlogov lahko pojavi pri funkciji [Izvajanje funkcije "readback"], vpliva na funkciji [ATC preverjanje] in [Prilagajanje vzletnim navodilom kontrole] ter posledično tudi na funkcijo [Izvajanje vzleta]. Torej resonanca od funkcije do funkcije poteka po povezavah **R1**, **R2.1**, **R2.2** in **R3**, ki so prikazane na sliki 4.3, od leve proti desni. Zakaj je povezava **R2** razčlenjena na dva dela **R2.1** in **R2.2** je pojasnjeno v razdelku 4.4.3.




**Slika 4.3** Slika, ki prikazuje kako so povezane funkcije [Interpretacija navodil zračne kontrole], [Izvajanje funkcije "readback"], [ATC preverjanje], [Prilagajanje vzletnim navodilom kontrole] in [Izvajanje vzleta]. Na njej so označene povezave R1, R2.1, R2.2 in R3, ki te funkcije povezujejo. Aspekti, označeni z rdečo barvo so tisti aspekti, ki so definirani.

## 4.4 Identifikacija protokola pri funkcijski resonanci

V pričujočem razdelku želimo razviti FRAM metodo v doslej še neraziskano smer. Metoda kot taka nam v tem trenutku ponuja podroben opis sistema oziroma sistemski model, ki ponazori kako sistem deluje na nivoju funkcij. S tem modelom si pomagamo pri zanesljivostni analizi opazovanega socio - tehničnega sistema. Iščemo torej vse možne scenarije, ki bi lahko povzročili neželjeni sistemski dogodek. FRAM metoda nam omogoči, da identificiramo potencialno tvegane povezave med aspekti funkcij, ne ponudi pa predpisanega postopka za nadaljnjo analizo teh povezav - zanima nas torej, kaj točno se prenaša po povezavah med aspekti funkcij in v kakšni obliki. Ali je morda možno tisto, kar se po opazovanih povezavah prenaša formalno zapisati in morebiti simulirati z računalniškim modelom? V tem razdelku poizkušamo razviti nadaljnje oziroma podrobnejše analitične

| Funkcija | Izhod funkcije | Variabilnost |
|---|---|---|
| [Interpretacija navodil zračne kontrole] | Interpretirana navodila zračne kontrole | Izhod iz te funkcije je nenatančen ampak pravočasen. |
| [Izvajanje funkcije "readback"] | Ponovljena navodila kontrole | Izhod funkcije je nenatančen oziroma nepravilen, izveden pa je pravočasno. |
| [ATC preverjanje] | Potrditev ali popravek | Izhod iz te funkcije je lahko navidezno pravilen, ampak pravočasen |
| [Prilagajanje vzletnim navodilom zračne kontrole] | Nahajamo se na vzletni stezi, obrnjeni v pravilno smer | Izhod funkcije je nenatančen oziroma nepravilen. Podan je lahko tudi prezgodaj. |
| [Izvajanje vzleta] | Letalo je v zraku | Izhod funkcije je nenatančen oziroma negotov, nedefiniran. Izveden je lahko prezgodaj. |

**Tabela 4.10** Analiza variabilnosti funkcij, prikazanih na sliki 4.3.

korake, ki bi nam morda lahko omogočali formalen zapis snovi oziroma podatkov, ki se po funkcijskih povezavah prenašajo. Za začetek v razdelkih 4.4.1 in 4.4.2 navedemo nekaj možnih scenarijev, ki povzročijo variabilnost izhoda opazovanih funkcij. Pri naštevanju teh scenarijev se torej nanašamo na sliko 4.3, kjer so označene opazovane tvegane povezave **R1**, **R2.1**, **R2.2** in **R3**. Čeprav oznaka **R1** pravzaprav definira dve povezavi, v tem primeru obe povezavi označimo s takšno oznako kot bi označili zgolj eno povezavo, saj se po obeh povezavah prenašajo identični podatki. Na povezavah **R1** se torej prenašajo podatki iz funkcije [Interpretacija navodil zračne kontrole] na aspekte funkcij [Izvajanje funkcije "readback"] ter [Prilagajanje navodilom zračne kontrole]. Ti podatki so lahko variabilni, ker je tudi izvajanje funkcije lahko variabilno - razumevanje oziroma interpretiranje podanih navodil je namreč kompleksna kognitivna naloga, pri kateri se marsikaj lahko zalomi. Komunikacijski nesporazumi so namreč neizogibni in se dogajajo vsakodnevno. Bolj pomembno je, kako te nesporazume rešujemo. Prav preprečevanju

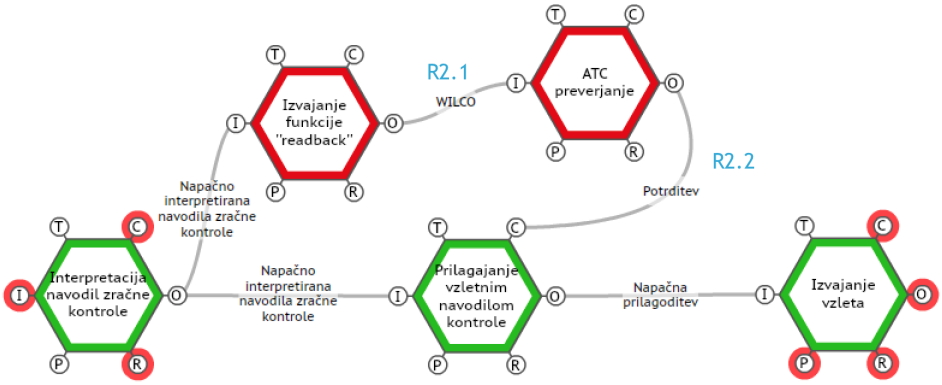
omenjenih nesporazumov sta namenjeni funkciji [Izvajanje funkcije "readback"] in [ATC preverjanje], ki pa nista odporni na napake. Tako je tudi pri funkciji [Prilagajanje vzletnim navodilom kontrole], saj je povezana s funkcijo [Izvajanje funkcije "readback"]. V nadaljevanju se osredotočamo na povezave **R2.1**, **R2.2** in **R3**, pri čemer se zavedamo, da nekatere napake oziroma neljubi dogodki, ki jih v nadaljevanju naštejemo, izvirajo tudi iz **neizogibne variabilnosti** izhoda funkcije [Interpretacija navodil zračne kontrole], ki pa potuje po obeh povezavah **R1**.

### 4.4.1 Identifikacija možne resonance na povezavah R2.1 in R2.2

Na povezavah **R2.1** in **R2.2** se prenaša izhod iz funkcije [Izvajanje funkcije "readback"] na aspekt vhod funkcije [ATC preverjanje], ki ima svoj izhod povezan na aspekt nadzor funkcije [Prilagajanje vzletnim navodilom kontrole]. Analize teh funkcij so predstavljene v razdelku 4.1. Kot je navedeno v tabeli 4.10, je lahko izhod iz funkcij [Izvajanje funkcije "readback"] in [ATC preverjanje] nenatančen oziroma nepravilen. Nekateri izmed scenarijev, v katerih je izhod iz teh dveh funkcij nepravilen, so sledeči:

- *napačna izvedba opazovanih funkcij zaradi slabe uporabe frazeologije*: Ta scenarij je omenjen že v razdelku 4.3 in sicer je omenjeno kršenje načel letalske frazeologije, ko pilot pomembna navodila ali dovoljenja kontrole potrdi z besedo "WILCO" (angl. *will comply*). Problem pri uporabi te fraze je, da kontrolor na ta način ne more vedeti, ali je pilot pravkar posredovano informacijo razumel pravilno, ali ne. Po viru [13] ugotovimo, da mora pilot vzletna navodila ali dovoljenje za vzlet eksplicitno ponoviti. Če kontrolor frazo "WILCO" vzame za zadostno, brez vedenja kaj je pilot dejansko razumel, lahko torej pride do neželjenega dogodka, kadar pilot narobe razume navodila, tako kontrolor kot tudi pilot pa sta prepričana, da so bila navodila razumljena pravilno in bodo tudi upoštevana. Ta scenarij je grafično prikazan z instanco opazovanega dela FRAM modela na sliki 4.4.

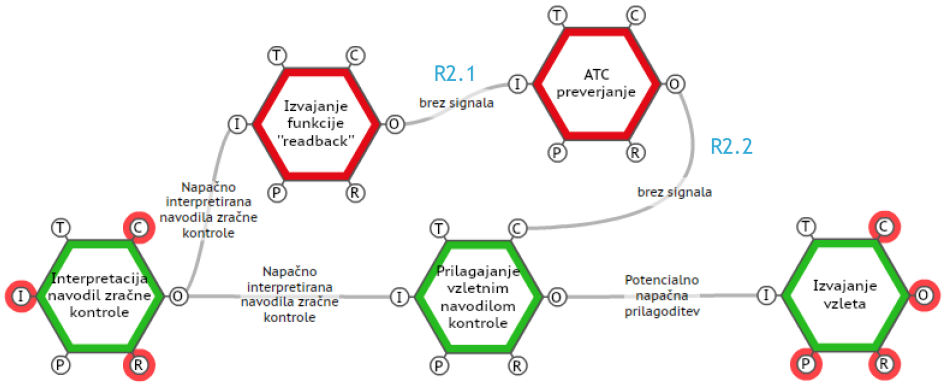
- *funkciji se sploh ne izvedeta, vendar se signal kljub temu pojavi na njunem izhodu*: Pilot v tem primeru enostavno misli, da se je funkcija izvedla pravilno in lahko nadaljuje z izvajanjem naslednje funkcije, ki za svojo izvedbo potrebuje aspekt, definiran z izhodom iz opazovane funkcije [ATC preverjanje]. Ta funkcija v modelu, grafično prikazanem na sliki 4.2, svoj izhod pelje še na funkcijo [Pridobitev dovoljenja za vzlet] poleg funkcije [Prilagajanje vzletnim navodilom kontrole]. To pomeni,




**Slika 4.4** Grafično prikazan scenarij *NAPAČNA IZVEDBA FUNKCIJE ZARADI SLABE UPORABE FRAZEOLOGIJE* na opazovanih povezavah R2.1 in R2.2. Aspekti, označeni z rdečo barvo so tisti aspekti, ki so definirani, funkciji pobarvani z rdečo barvo pa sta opazovani funkciji.

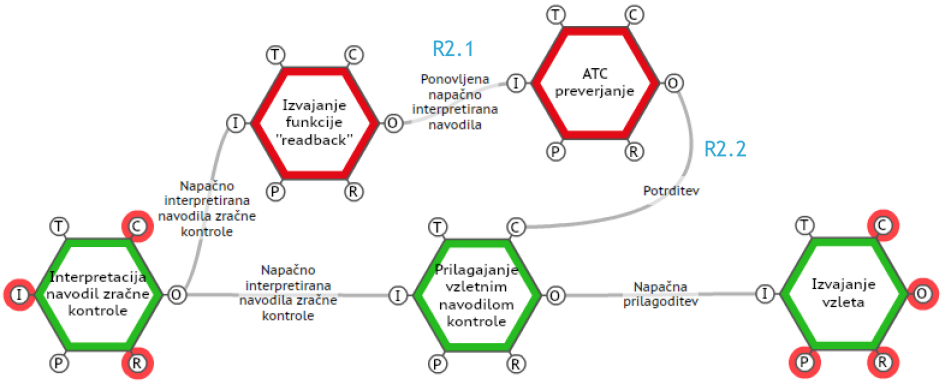
da bo variabilnost izhoda vplivala na izvedbo obeh navedenih funkcij, ki se izvajata istočasno in sta obe nujno potrebni za uspešno izvedbo končne funkcije [Izvajanje vzleta], funkcija [Pridobitev dovoljenja za vzlet] pa kljub temu za nadaljnjo analizo ne bo prišla v poštev v pričujočem delu. V viru [14] lahko preberemo primer iz leta 2012, ko je komercialno letalo Boeing 737-800 na nizozemskem letališču Eindhoven vzletelo brez dovoljenja zračne kontrole. Ta specifičen primer je sicer bolj zapleten, kot prikazuje naš model, saj je tudi dano letalo mnogo bolj kompleksno kot tisto, ki ga opisuje postavljeni model, vendar če ta primer poenostavimo, ga lahko opišemo z definiranimi funkcijami. Pilot oziroma v tem primeru kar oba pilota letala Boeing 747-800 sta predvidevala, da sta uspešno izvedla tako funkciji [Interpretacija navodil zračne kontrole], [Izvajanje funkcije ”readback”], kot tudi funkciji [Prilagajanje vzletnim navodilom kontrole] ter [Pridobitev dovoljenja za vzlet], čeprav se v resnici funkcije [Izvajanje funkcije ”readback”], [ATC preverjanje] in [Pridobitev dovoljenja za vzlet] sploh niso izvedle, funkciji [Prilagajanje vzletnim navodilom kontrole] in [Interpretacija navodil zračne kontrole] pa sta se izvedli, vendar nepravilno. K takemu rezultatu je močno prispevalo slabo razumevanje navodil letališke kontrole. Ta scenarij je grafično prikazan z instanco opazovanega dela FRAM modela na sliki 4.5.

- *kontrolor potrdi nepravilen ”readback”*: Pilot torej interpretira navodila napačno in to napako pri izvajanju funkcije ”readback” tudi pove. Kontrolor v tem scenariju




**Slika 4.5** Grafično prikazan scenarij *FUNKCIJI SE SPLOH NE IZVEDETA, VENDAR SE SIGNAL KLJUB TEMU POJAVI NA IZHODU* z opazovanima povezavama R2.1 in R2.2. Na povezavah R2.1 in R2.2 se namenski signal v resnici ne pojavi, opis podatkov "brez signala"lahko v tem primeru interpretiramo tudi kot lažni signal. Aspekti, označeni z rdečo barvo so tisti aspekti, ki so definirani, funkcije pobarvane z rdečo barvo pa so opazovane funkcije.

to napako presliši oziroma predvideva, da je bil "readback" izveden pravilno in s tem pusti pilotu, da nadaljuje izvajanje svojih funkcij. To lahko vodi do neljubega dogodka, odvisno od tega pri katerem podatku se je napaka zgodila - lahko je to smer steze, lahko vrednost, ki jo mora pilot vnesti v transponder in podobno, v vsakem primeru pa takšna napaka predstavlja tveganje. Opisana situacija je z instanco opazovane sekcije FRAM modela prikazana na sliki 4.6.

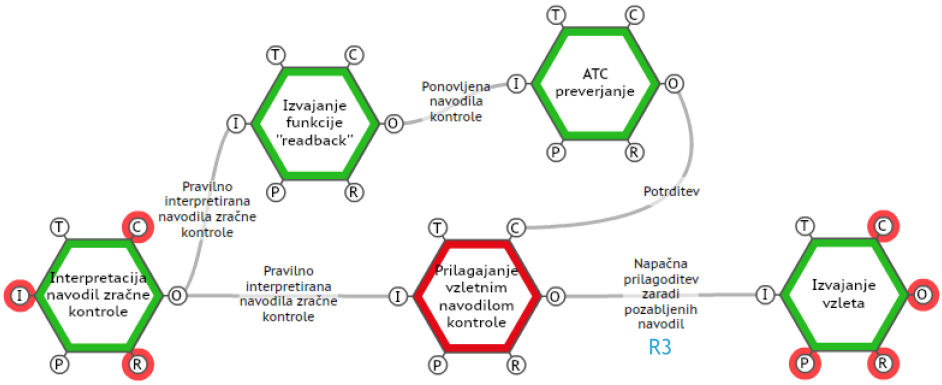



**Slika 4.6** Grafično prikazan scenarij *KONTROLOR POTRDI NEPRAVILEN "READBACK"* z opazovanima povezavama R2.1 in R2.2 Aspekti, označeni z rdečo barvo so tisti aspekti, ki so definirani, funkcije pobarvane z rdečo barvo pa so opazovane funkcije.

### 4.4.2 Identifikacija možne resonance na povezavi R3

Na povezavi **R3** se prenaša izhod iz funkcije [Prilagajanje vzletnim navodilom zračne kontrole] na aspekt vhod funkcije [Izvajanje vzleta]. Neželjeni scenariji, ki se lahko zgodijo pri izvajanju funkcije [Prilagajanje vzletnim navodilom kontrole], so sledeči:

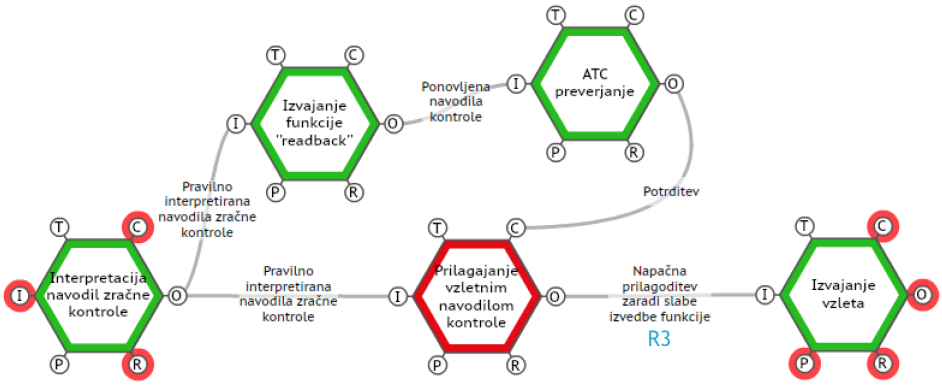
- *pilot pozabi upoštevati dogovorjena navodila*: Funkcije [Interpretacija navodil zračne kontrole], [Izvajanje funkcije "readback"] in [ATC preverjanje] so se izvedle brez napak, vendar se pilot kljub temu nepravilno prilagodi dogovorjenim navodilom. Resonanca iz prejšnjih povezav torej v tem primeru ni relevantna, funkcija [Prilagajanje vzletnim navodilom kontrole] pa na svoj izhod vseeno pošlje signal, ki ni pravilen, torej se letalo ne drži navodil, dogovorjenih s kontrolo letenja, zaradi napake pilota, ki ima v glavi drugačna navodila od dogovorjenih. Izhod iz funkcije je torej nenatančen oziroma nepravilen. Opisana situacija je z instanco analizirane sekcije FRAM modela prikazana na sliki 4.7.




**Slika 4.7** Prikazan je scenarij *PILOT POZABI UPOŠTEVATI DOGOVORJENA NAVODILA* z opazovano povezavo R3. Aspekti, označeni z rdečo barvo so tisti aspekti, ki so definirani, funkcija pobarvana z rdečo barvo pa je opazovana funkcija.

- *pilot nehote krši dogovorjena navodila*: Pilot lahko zaradi različnih razlogov krši dogovorjena navodila, kar lahko pripelje do neželjenega dogodka. Včasih se lahko zgodi napaka, pri kateri je pilot prepričan, da sledi danim navodilom zračne kontrole in pri tem nehote uporabi npr. napačno smer steze pri izvajanju vzleta ali pa zavije na napačno *taxiway* stezo pri vožnji po tleh. V viru [15] najdemo primer, ki

je na kratko analiziran s pomočjo FRAM modela. To je letalska nesreča komercialnega leta Comair 5191, ko je letalo CRJ 100ER na letališču Lexington Blue Grass Airport v ZDA uporabilo napačno vzletno stezo[3], ki je bila za zmogljivosti tega letala prekratka in posledično je letalo strmoglavilo pri poizkusu vzletanja in pri tem s seboj vzelo 49 življenj. Podrobnosti in vzroki nesreče so na voljo v viru [15], primer pa je tukaj kot dokaz, da se takšne napake res dogajajo. Takšna napaka je sicer lahko zgolj posledica variabilnosti izvajanja funkcije [Prilagajanje vzletnim navodilom kontrole], vendar taki napaki mnogokrat botruje več dejavnikov, ki preko resonance variabilnost te funkcije še povečajo in s tem močno povišajo možnosti za tovrstni scenarij. Ne glede na razlog za takšno izvedbo funkcije [Prilagajanje vzletnim navodilom kontrole], se na povezavi **R3** nahaja nepravilen oziroma napačen izhod oziroma lažni pravilni izhod (angl. *false positive*). Opisana situacija je z instanco analizirane sekcije FRAM modela prikazana na sliki 4.8.

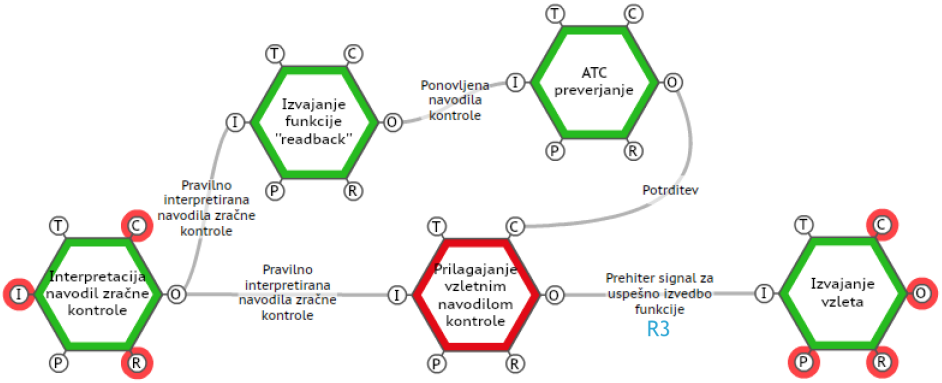



**Slika 4.8** Prikazan je scenarij *PILOT NEHOTE KRŠI DOGOVORJENA NAVODILA* z opazovano povezavo R3. Aspekti, označeni z rdečo barvo so tisti aspekti, ki so definirani, funkcija pobarvana z rdečo barvo pa je opazovana funkcija.

- *funkcija [Prilagajanje navodilom zračne kontrole] se sploh ne izvede ali pa se izvede samo delno*: Obstaja možnost, da funkcija [Prilagajanje navodilom zračne kontrole] na svoj izhod in s tem na povezavo **R3** postavi signal, s katerim sporoča, da je bila uspešno opravljena, čeprav v resnici sploh ni bila izvedena, ali pa je bila izvedena samo delno. Enega izmed takih primerov najdemo v viru [16] in sicer je govora o incidentu, ki se je zgodil 26. novembra 2010 na hongkongškem letališču, ko je letalo Airbus A340-300 finske letalske družbe Finnair poizkusilo vzleteti kar iz *taxiway*

[3]Mnoga letališča imajo več kot eno vzletno-pristajalno stezo

steze, ki je namenjena vožnji letal po tleh. Pri incidentu sicer ni nastala nobena škoda, razen pregretja zavor letala, je pa ta incident opomin na dejstvo, kako zelo hitro se lahko zgodi takšna napaka. Če ta incident površno in grobo prevedemo na naš FRAM model lahko trdimo, da funkcija [Prilagajanje navodilom zračne kontrole] sploh ni bila izvedena, na izhodu funkcije pa se je vseeno pojavil signal, ki trdi, da je bila funkcija uspešno izvedena (angl. *false positive*). Tudi tak scenarij se navadno težko zgodi "sam od sebe", bolj verjetno pa, če je prisotna resonanca iz predhodnih funkcij. Opisana situacija je z instanco analizirane sekcije FRAM modela prikazana na sliki 4.9.




**Slika 4.9** Prikazan je scenarij *FUNKCIJA [Prilagajanje navodilom zračne kontrole] SE SPLOH NE IZVEDE, ALI PA SE IZVEDE SAMO DELNO* z opazovano povezavo R3. Aspekti, označeni z rdečo barvo so tisti aspekti, ki so definirani, funkcija pobarvana z rdečo barvo pa je opazovana funkcija.

Pri zadnjih dveh naštetih primerih so razlike manj očitne kot pri povezavah **R2.1** in **R2.2**. Zlasti zadnji scenarij bi lahko pojasnili tudi z razlago, da *pilot nehote krši dogovorjena pravila*. Razlika med obema razlagama je ta, da se pri prvi, torej razlagi *pilot nehote krši dogovorjena pravila* funkcija [Prilagajanje navodilom zračne kontrole] izvaja, vendar njen izhod "pokvari" variabilnost izvajanja te funkcije, pri drugi razlagi pa se funkcija [Prilagajanje navodilom zračne kontrole] sploh ne izvede, ali pa se izvede samo delno.

### 4.4.3 Opis prenašanih podatkov

V tem razdelku skušamo formalno opisati podatke, ki v scenarijih, opisanih v razdelkih 4.4.1 in 4.4.2, potujejo po povezavah **R1**, **R2.1**, **R2.2** in **R3**.

### Podatki na povezavi R1

Povezava **R1** je na sliki 4.3 in vseh njenih opazovanih instancah vsebovana dvakrat - obe povezavi sta označeni z isto oznako, ker se po obeh prenaša identičen podatek. Povezavi prenašata izhod funkcije [Interpretacija navodil zračne kontrole] na vhod funkcij [Izvajanje funkcije "readback"] in [Prilagajanje vzletnim navodilom kontrole]. Izhod funkcije [Interpretacija navodil zračne kontrole] je v razdelku 4.1.4 definiran kot **interpretirana navodila zračne kontrole**. Po povezavah **R1** torej potuje neka informacija, ki je **rezultat kognitivnega procesa** pilota. Gre za podatke, po katerih se pilot v praksi ravna in so preverjeni preko funkcije [ATC preverjanje]. Vzemimo takšne podatke iz scenarija *funkciji se sploh ne izvedeta, vendar se signal kljub temu pojavi na izhodu*, ki je opisan v razdelku 4.4.1, kjer sta opazovani povezavi **R2.1** in **R2.2**. Na sliki 4.10 lahko vidimo primer komunikacije med pilotom (P2) in kontrolo letenja (GND[4]). Kognitivni proces, oziroma izvedba funkcije [Interpretacija navodil zračne kontrole], za vhod vzame informacije, ki jih je v tem dialogu podal kontrolor (GND), na izhod pa vrne interpretacijo le-teh. Po povezavi **R1** se torej prenaša podatkovna struktura (ta se sicer v tem trenutku nahaja zgolj v glavi pilota), ki v tem primeru vsebuje sledeče podatke:

- klicni znak letala (angl. *callsign*),
- določeno pot do cilja, ki vsebuje dogovorjene navigacijske točke,
- določeno višino leta,
- določeno kodo transponderja.

| | |
|---|---|
| GND | RA 1YG, cleared to destination LONDON STANSTED via RAPSO 2 JULIET departure, and initially FL 060, squawk 7342 |
| P2 | cleared to destination LONDON STANSTED on . . . . unclear) departure, initially climb and maintain FL 60 to squawk 7342, RA 1YG |
| GND | RA 1YG, read back correct |

**Slika 4.10** Komunikacija med pilotom P2 in letališko kontrolo GND, povzeta po viru [14].

[4]Kontrola letenja se na mnogih letališčih deli na dva dela, ki delujeta na različnih radijskih frekvencah. En del je dodeljen zračnim operacijam, drugi pa talnim.

#### Podatki na povezavah R2.1 in R2.2

Odgovor pilota v dialogu, prikazanem na sliki 4.10 predstavlja izvedbo funkcije [Izvajanje funkcije "readback"]. Ta funkcija na izhod pošlje signal, ki po povezavi **R2.1** potuje v obliki **radijskih valov**, ki prenašajo ponovljena navodila zračne kontrole do vhoda funkcije [ATC preverjanje]. Podatkovna struktura je pri tem sestavljena na isti način kot tista, ki se prenaša po povezavi **R1**.

Funkcija [ATC preverjanje] na svoj izhod da signal, ki prav tako v obliki radijskih valov potuje do aspekta **nadzor** funkcije [Prilagajanje vzletnim navodilom kontrole], vendar podatkovna struktura vsebuje manj podatkov - potrditev ali popravek napake. Povezavi **R2.1** in **R2.2** sta torej označeni na tak način zato, ker se po teh dveh povezavah prenašajo **podatki istega tipa**, ne pa identični podatki, tako kot po povezavah **R1**.

#### Podatki na povezavi R3

Povezava **R3** povezuje funkciji [Prilagajanje vzletnim navodilom kontrole] in [Izvajanje vzleta]. Podatki, ki se po tej povezavi prenašajo, so v razdelku 4.1.8 definirani kot sporočilo "nahajamo se na vzletni stezi, obrnjeni v pravilno smer". Ta trditev je torej lahko pravilna ali napačna. Tako funkcija [Izvajanje vzleta] iz te povezave potrebuje zgolj vrednosti **TRUE** in **FALSE**, ki pa so lahko pravilne ali pa navidezno pravilne (angl. *false positive*) in navidezno napačne (angl. *false negative*).

### 4.4.4 Prikaz resonance prenašanih podatkov skozi primer

V tem razdelku s pomočjo scenarija *funkciji se sploh ne izvedeta, vendar se signal kljub temu pojavi na izhodu funkcije* pojasnimo, kakšna je variabilnost podatkov, ki so opisani v razdelku 4.4.3. Na sliki 4.11 je prikazan najbolj kritičen del komunikacije med pilotoma in kontrolo, ki je pripeljala do incidenta. Iz slike 4.11 je razvidno, da sta pilota, skupaj z informacijami pridobljenimi prej (prikazano na sliki 4.10), **napačno interpretirala navodila zračne kontrole**. Kot posledica te napačne interpretacije se funkcija [Pridobitev dovoljenja za vzlet] (ki sicer ni fokus naše analize) sploh ni izvedla, prav tako pa se nista izvedli funkciji [Izvajanje funkcije "readback"] in [ATC preverjanje], ki poskrbita za aspekt **nadzor** pri funkciji [Pridobitev dovoljenja za vzlet]. Na povezavi **R1** se torej pojavijo napačno interpretirana navodila zračne kontrole - v razdelku 4.4.3 opisana podatkovna struktura vsebuje **navidezno dovoljenje za vzlet**. Takšna nepravilna podatkovna struktura na izvajanje nadaljnjih funkcij vpliva v tej meri, da se te sploh ne

| | |
|---|---|
| GND | 1YG, this one to right, and for departure contact tower, 131.0, good flight, bye bye |
| P2 | 121.0, thank you for your help sir, RA 1YG |
| P1 ic | and . . the retracts . . . . we are clear takeoff, aren't we ? |
| P2 ic | we are cleared takeoff yeah . . . . it's after airborne contact tower in the air . . . I think that was it |
| P2 ic | OK, retracts and checks complete, we're cleared takeoff |
| P1 ic | OK |

**Slika 4.11** Notranja komunikacija med pilotoma P1 in P2 (označena s črkama "ic") in komunikacija med letalom in letališko kontrolo, povzeta po viru [14].

izvedejo. Na povezavah **R2.1** in **R2.2** se torej ne dogaja nič, vseeno pa se na izhodu funkcije [Prilagajanje vzletnim navodilom kontrole], torej na povezavi **R3** pojavi signal, ki dovoljuje izvedbo funkcije [Izvajanje vzleta], torej navidezno pravilen signal.

# 5 Zaključek

V diplomskem delu smo opredelili, kaj je to socio - tehnični sistem in poiskali tipičen primer tovrstnega sistema. Pod drobnogled smo vzeli pilotiranje enomotornega športnega letala. Ugotovili smo, da je ta socio - tehnični sistem zelo kompleksen in primeren za podrobnejšo analizo, saj se v tem sistemu kljub dovršenosti uporabljane tehnike in natančno predpisanim operativnim proceduram pogosto dogajajo incidenti, ki jim marsikdaj botruje človeška napaka. Ta sistem smo tudi uspešno in s primerno natančnostjo za pričujoče delo opisali. Podrobnosti opravljanja pilotskih nalog v manjšem enomotornem športnem letalu v tem delu bazirajo na podlagi avtorjevih lastnih izkušenj in znanja.

V poglavju 3 smo opisali glavna načela FRAM metode, s katero smo opisani socio - tehnični sistem tudi analizirali. V poglavju 4 smo glede na opisani socio - tehnični sistem poizkušali identificirati sistemske funkcije in jih povezati v FRAM model na tak način, da bi čim bolj ustrezale opisanemu delovanju sistema. Pri tem smo sledili predpisanim načelom FRAM metode. Nato smo identificirali variabilnost funkcij in njeno agregacijo oziroma funkcijsko resonanco na postavljenem modelu. S tem smo izvedli korake FRAM metode, vendar zgolj do vključno koraka številka tri. Četrtega, oziroma zadnjega koraka,

torej razvoja priporočil za omejevanje variabilnosti problematičnih sistemskih funkcij, v pričujočem delu nismo izvedli. Kot primer izvedbe tega koraka, lahko pri izvedbi funkcije [Interpretacija navodil zračne kontrole] predlagamo dodatno nalogo in sicer zapisovanje navodil zračne kontrole. Tako pilot, namesto da ima navodila samo v spominu, le-ta prenese na list papirja in s tem dejansko zmanjša variabilnost funkcije [Interpretacija navodil zračne kontrole]. Zapisovanje navodil kontrole na papir sicer prakticira mnogo pilotov, še posebno, če so dana navodila kontrole dolga.

Namesto razvijanja priporočil smo se odločili, da bomo poizkušali metodo formalno razviti v do sedaj še neraziskano smer. Poizkušali smo namreč raziskati, kaj točno se prenaša po povezavah med funkcijami opazovanega sistema, da bi lahko takšen teoretično opisan FRAM model morda v prihodnosti simulirali z računalniškim modelom. S tem smo prišli do ugotovitve, da se v našem definiranem FRAM modelu po opazovanih povezavah prenašajo podatkovne strukture, ki so lahko v obliki miselnega procesa ali pa radijskih valov, lahko pa se po teh povezavah prenašajo zgolj enostavne vrednosti tipa true in false.

Razvoj FRAM metode bi se lahko nadaljeval tako, da bi še bolj formalno definirali podatkovne strukture, ki po povezavah FRAM modela potujejo, ter da bi to storili za cel model, ne pa zgolj na določeni sekciji modela. Tako bi naredili korak bližje k računalniški simulaciji modelov socio - tehničnih sistemov, razvitih s FRAM metodo. Dolgoročni cilj izvajanja analiz sodobnih kompleksnih socio - tehničnih sistemov je namreč ta, da bi se te izvajale zgolj proaktivno, ne pa tudi reaktivno. Dolgoročno torej želimo vse neželjene dogodke s temeljitimi zanesljivostnimi analizami predvideti in jih preprečiti, še preden se zgodijo. Metoda FRAM bo morda v prihodnosti lahko močno prispevala k temu cilju.

# Literatura


[1] G. H. Walker, N. A. Stanton, P. M. Salmon, D. P. Jenkins, A review of sociotechnical systems theory: A classic concept for new command and control paradigms, Theoretical Issues in Ergonomics Science 9 (6) (2008) 479–499.

[2] E. Hollnagel, FRAM, The Functional Resonance Analysis Method: Modelling Complex Socio-technical Systems, Ashgate Publishing, Ltd., 2012.

[3] Vizzlo, https://vizzlo.com/, (čas dostopa: julij 2017).

[4] Traffic pattern description, https://www.ivao.aero/training/documentation/books/PP_ADC_Traffic_Pattern_Description.pdf, (čas dostopa: september 2017).

[5] N. Leveson, Engineering a safer world: Systems thinking applied to safety, MIT Press, 2011.

[6] S. Dekker, The field guide to human error, http://www.leonardo-in-flight.nl/PDF/FieldGuide%20to%20Human%20Error.PDF, (čas dostopa: julij 2017).

[7] Z. H. Qureshi, M. A. Ashraf, Y. Amer, Modeling industrial safety: A sociotechnical systems perspective, in: Industrial Engineering and Engineering Management, 2007 IEEE International Conference on, IEEE, 2007, pp. 1883–1887.

[8] Z. H. Qureshi, A review of accident modelling approaches for complex socio-technical systems, in: Proceedings of the Twelfth Australian workshop on Safety critical systems and software and safety-related programmable systems - Volume 86, Australian Computer Society, Inc., 2007.

[9] T. Ishimatsu, N. G. Leveson, J. Thomas, M. Katahira, Y. Miyamoto, H. Nakao, Modeling and hazard analysis using STPA, Proceedings of the 4th IAASS Con-

ference, Making Safety Matter, https://dspace.mit.edu/handle/1721.1/79639, (čas dostopa: julij 2017).

[10] E. Hollnagel, J. Hounsgaard, L. Colligan, C. for Kvalitet (Region Syddanmark), FRAM - the Functional Resonance Analysis Method: A Handbook for the Practical Use of the Method, Centre for Quality, Region of Southern Denmark, 2014, http://functionalresonance.com/onewebmedia/FRAM_handbook_web-2.pdf, (čas dostopa: julij 2017).

[11] FRAM Model Visualiser, http://functionalresonance.com/FMV/index.html, (čas dostopa: julij 2017).

[12] AIP Slovenia, https://www.sloveniacontrol.si/acrobat/aip/Operations/2013-05-10/pdf/LJ-AD-2.LJLJ-en-GB.pdf, (čas dostopa: julij 2017).

[13] K. Štumberger, Priročnik o uporabi frazeologije v slovenskem jeziku, http://www.caa.si/fileadmin/user_upload/pageuploads/Prirocnik_o_uporabi_frazeologije_FRAZEO_slo_pdf.pdf, (čas dostopa: avgust 2017).

[14] Dutch Safety Board, Take off without permission, Boeing 737-800, 11 October 2012 Eindhoven Airport, https://www.onderzoeksraad.nl/en/onderzoek/1919/take-off-without-permission-boeing-737-800-11-october-2012, (čas dostopa: avgust 2017).

[15] E. Hollnagel, S. Pruchnicki, R. Woltjer, S. Etcher, Analysis of Comair flight 5191 with the functional resonance accident model, in: 8th International Symposium of the Australian Aviation Psychology Association, 2008.

[16] Accident Investigation Division, Civil Aviation Department, Hong Kong Special Administrative Region, Report on the Serious Incident of an Attempted Take-off on Taxiway Flight FIN070 Airbus A340-300 OH-LQD at the Hong Kong International Airport on 26 November 2010 [27 November 2010 Local Time], http://www.cad.gov.hk/reports/B-LAT1-2011.pdf, (čas dostopa: avgust 2017).